L Number	Hits	Search Text	DB	Time stamp
1	22254	network near4 (manag$7 monitor$3)	USPAT	2004/03/19 17:18
2	41128	(chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device terminal computer workstation icon line link)	USPAT	2004/03/19 17:20
3	41621	(chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)	USPAT	2004/03/19 17:21
4	604	(network near4 (manag$7 monitor$3)) and ((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link))	USPAT	2004/03/19 17:21
5	988	(network near4 (manag$7 monitor$3)) and metric	USPAT	2004/03/19 17:22
6	2367	metric same (link flow status traffic)	USPAT	2004/03/19 17:23
7	371	(network near4 (manag$7 monitor$3)) and (metric same (link flow status traffic))	USPAT	2004/03/19 17:23
8	14263	color adj1 cod$3	USPAT	2004/03/19 17:23
9	446344	red green yellow orange	USPAT	2004/03/19 17:24
10	484325	((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange)	USPAT	2004/03/19 17:24
11	3411	(network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))	USPAT	2004/03/19 17:25
12	98	((network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))) and (metric same (link flow status traffic))	USPAT	2004/03/19 17:25
13	92	(((network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))) and (metric same (link flow status traffic))) and (memrory database view)	USPAT	2004/03/19 17:27
14	11503	network with (topolog$4 map graph)	USPAT	2004/03/19 17:27
15	51	((((network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))) and (metric same (link flow status traffic))) and (memrory database view)) and (network with (topolog$4 map graph))	USPAT	2004/03/19 17:35

16	41	((((network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))) and (metric same (link flow status traffic))) and (memrory database view)) not (((((network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))) and (metric same (link flow status traffic))) and (memrory database view)) and (network with (topolog$4 map graph)))	USPAT	2004/03/19 17:46
17	118	((network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))) and (network with (topolog$4 map graph)) and metric$2	USPAT	2004/03/19 17:47
18	67	(((network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))) and (network with (topolog$4 map graph)) and metric$2) not (((((network near4 (manag$7 monitor$3)) and (((chang$3 alter$3 modif$7) with (size shape color appearance thick$4 bold$4) with (device node terminal computer workstation icon line link)) (color adj1 cod$3) (red green yellow orange))) and (metric same (link flow status traffic))) and (memrory database view)))	USPAT	2004/03/19 17:47

L Number	Hits	Search Text	DB	Time stamp
1	11794	network near3 (topolog$4 graph)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:30
2	27270	network with (link line) with connect$3 with (device terminal computer terminal)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 13:21
3	18956	chang$3 with (size shape color appearance thickness) with (line link)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:49
4	150	(network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:32
5	7	((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))) and metric	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
6	56363	chang$3 with (size shape color appearance thickness) with (device terminal line link)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
7	355	(network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
8	16	((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))) and metric	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
9	9	(((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))) and metric) not (((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))) and metric)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
10	73406	(alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:55
11	547	(network with (link line) with connect$3 with (device terminal computer terminal)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:55
12	26	((network with (link line) with connect$3 with (device terminal computer terminal)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link))) and metric$5	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 14:44

13	10	(((network with (link line) with connect$3 with (device terminal computer terminal)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link))) and metric$5) not ((((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))) and metric) (((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))) and metric))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:56
14	278939	(link line) with connect$3 with (device terminal computer terminal)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 13:22
15	84	(network near3 (topolog$4 graph)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and ((link line) with connect$3 with (device terminal computer terminal))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 14:40
16	15	((network near3 (topolog$4 graph)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and ((link line) with connect$3 with (device terminal computer terminal))) and metric	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 13:22
17	1	("6639893").PN.	USPAT	2004/03/19 14:40
18	0	(("6639893").PN.) and (((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))) and metric)	USPAT	2004/03/19 14:41
19	0	(("6639893").PN.) and ((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link)))	USPAT	2004/03/19 14:41
20	1	(("6639893").PN.) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))	USPAT	2004/03/19 14:42
21	1	((("6639893").PN.) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))) and metric	USPAT	2004/03/19 14:42
22	1139	(network near3 (topolog$4 graph)) and metric$5	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 14:44
23	93	((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and ((network near3 (topolog$4 graph)) and metric$5)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 14:45

L Number	Hits	Search Text	DB	Time stamp
1	11794	network near3 (topolog$4 graph)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:30
2	27270	network with (link line) with connect$3 with (device terminal computer terminal)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 13:21
3	18956	chang$3 with (size shape color appearance thickness) with (line link)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:49
4	150	(network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:32
5	7	((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))) and metric	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
6	56363	chang$3 with (size shape color appearance thickness) with (device terminal line link)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
7	355	(network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
8	16	((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))) and metric	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
9	9	(((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))) and metric) not (((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))) and metric)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:37
10	73406	(alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 15:37
11	547	(network with (link line) with connect$3 with (device terminal computer terminal)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:55
12	26	((network with (link line) with connect$3 with (device terminal computer terminal)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link))) and metric$5	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 14:44

13	10	(((network with (link line) with connect$3 with (device terminal computer terminal)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link))) and metric$5) not ((((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))) and metric) (((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))) and metric))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 12:56
14	278939	(link line) with connect$3 with (device terminal computer terminal)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 13:22
15	84	(network near3 (topolog$4 graph)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and ((link line) with connect$3 with (device terminal computer terminal))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 14:40
16	15	((network near3 (topolog$4 graph)) and ((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and ((link line) with connect$3 with (device terminal computer terminal))) and metric	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 13:22
17	1	("6639893").PN.	USPAT	2004/03/19 14:40
18	0	(("6639893").PN.) and (((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link))) and metric)	USPAT	2004/03/19 14:41
19	0	(("6639893").PN.) and ((network with (link line) with connect$3 with (device terminal computer terminal)) and (chang$3 with (size shape color appearance thickness) with (line link)))	USPAT	2004/03/19 14:41
20	1	(("6639893").PN.) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))	USPAT	2004/03/19 14:42
21	1	((("6639893").PN.) and (chang$3 with (size shape color appearance thickness) with (device terminal line link))) and metric	USPAT	2004/03/19 14:42
22	1139	(network near3 (topolog$4 graph)) and metric$5	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 14:44
23	93	((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and ((network near3 (topolog$4 graph)) and metric$5)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 14:59
24	764	((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and metric	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 15:00
25	71	((link line) with connect$3 with (device terminal computer terminal)) and (((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and metric)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 15:07

27	7	((alter$3 updat$3 modif$7 chang$3) with (size shape dark$4 bold$4 color appearance thickness) with (line device symbol terminal link)) and ((345/736).CCLS.)	USPAT	2004/03/19 15:29
28	1	("6496209").PN.	USPAT	2004/03/19 15:29
29	4915	(network near3 (topolog$4 graph)) and (memory stor$3 metric)	USPAT	2004/03/19 15:45
30	1	(("6496209").PN.) and (memory stor$3 metric)	USPAT	2004/03/19 15:30
26	65	(345/736).CCLS.	USPAT	2004/03/19 15:34
31	29	((345/736).CCLS.) and ((size shape dark$4 bold$4 color appearance thickness) with (line device connector symbol terminal node link))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/19 15:45
32	1	("6535227").PN.	USPAT	2004/03/19 15:45
33	1	(("6535227").PN.) and (memory stor$3 metric)	USPAT	2004/03/19 16:11
34	1	(assessing with security with posture with network).ti.	USPAT	2004/03/19 16:12
35	1	((assessing with security with posture with network).ti.) and metric	USPAT	2004/03/19 16:12

(17)

US006691256B1

(12) United States Patent
Cook et al.

(10) **Patent No.: US 6,691,256 B1**
(45) **Date of Patent: Feb. 10, 2004**

(54) **NETWORK PROBLEM INDICATION**

(75) Inventors: **Mark Douglas Cook**, St Albans (GB); **Lee Anthony Walker**, Watford (GB); **Simon Peter Valentine**, Hemel Hempstead (GB); **Russell Kennett Bulmer**, Hemel Hempstead (GB)

(73) Assignee: **3Com Corporation**, Santa Clara, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/527,635**

(22) Filed: **Mar. 17, 2000**

(30) **Foreign Application Priority Data**

Jun. 10, 1999 (GB) 9913531

(51) **Int. Cl.**^{7} **G06F 11/00**; H04L 1/22
(52) **U.S. Cl.** **714/43**; 709/224
(58) **Field of Search** 714/43, 47, 48, 714/57, 28, 30, 31, 37, 39, 44, 46, 38; 709/223, 224

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,377,196 A	*	12/1994	Godlew et al.	371/20.1
5,436,909 A		7/1995	Dev et al.	371/20.1
5,471,399 A		11/1995	Tanaka et al.	364/491
5,748,880 A	*	5/1998	Ito et al.	395/183.22
5,913,036 A	*	6/1999	Brownmiller et al.	395/200.54
6,006,016 A	*	12/1999	Faigon et al.	395/185.01
6,173,422 B1	*	1/2001	Kimura et al.	714/57
6,178,531 B1	*	1/2001	Kolb	714/715
6,269,401 B1	*	7/2001	Fletcher et al.	709/224
6,327,677 B1	*	12/2001	Garg et al.	714/37

FOREIGN PATENT DOCUMENTS

EP	0 810 756 A2	12/1997	H04L/12/24
GB	2 286 317 A	8/1995	H04L/12/26
GB	2 328 043 A	2/1999	G06F/17/30
WO	WO 97/35409	9/1997	H04L/12/56
WO	WO 98/25377	6/1998	H04L/12/24

* cited by examiner

Primary Examiner—Nadeem Iqbal
(74) *Attorney, Agent, or Firm*—McDonnell Boehnen Hulbert & Berghoff

(57) **ABSTRACT**

Network supervising apparatus, computer program, and method of supervising a network comprising:

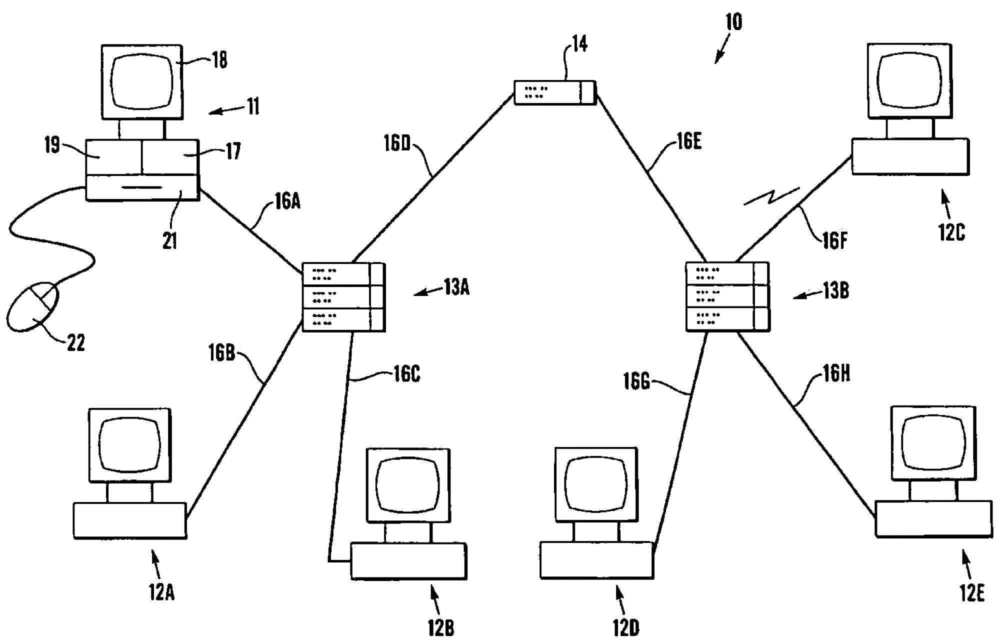
applying an algorithm to information relating to the devices of the network to provide a stress value,

comparing the stress value with a predetermined limit, retrieving graphic symbol signals from a signal store and providing the graphic symbol on a visual display apparatus when the stress value reaches the predetermined limit,

manually selecting the graphic symbol and causing said visual display apparatus to provide an image indicating where the stress value has reached the predetermined limit.

Stress value

57 Claims, 2 Drawing Sheets



Metric indicator

03/19/2004, EAST Version: 1.4.1

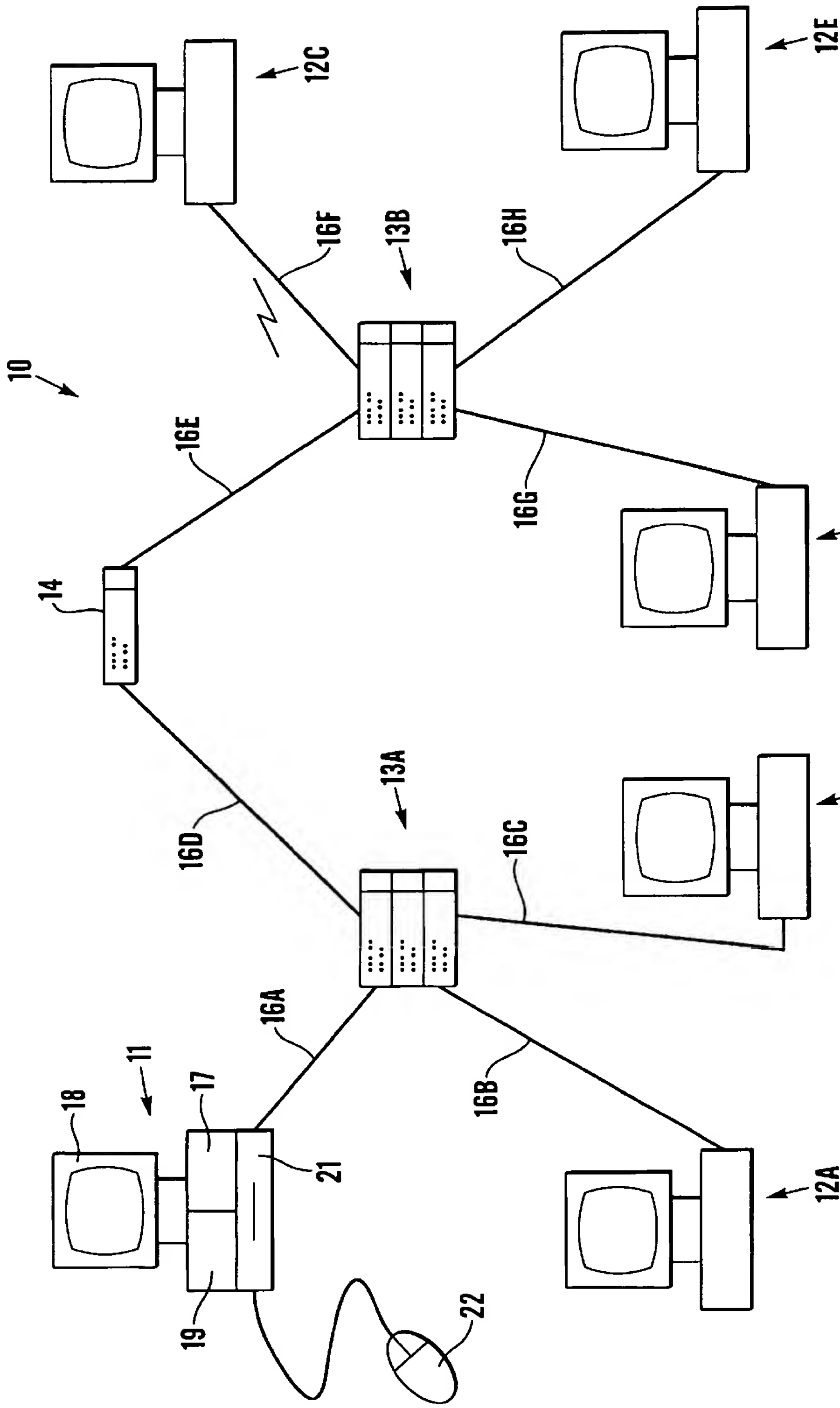

10
11
12A
12B
12C
12D
12E
13A
13B
14
16A
16B
16C
16D
16E
16F
16G
16H
17
18
19
21
22

Fig. 1

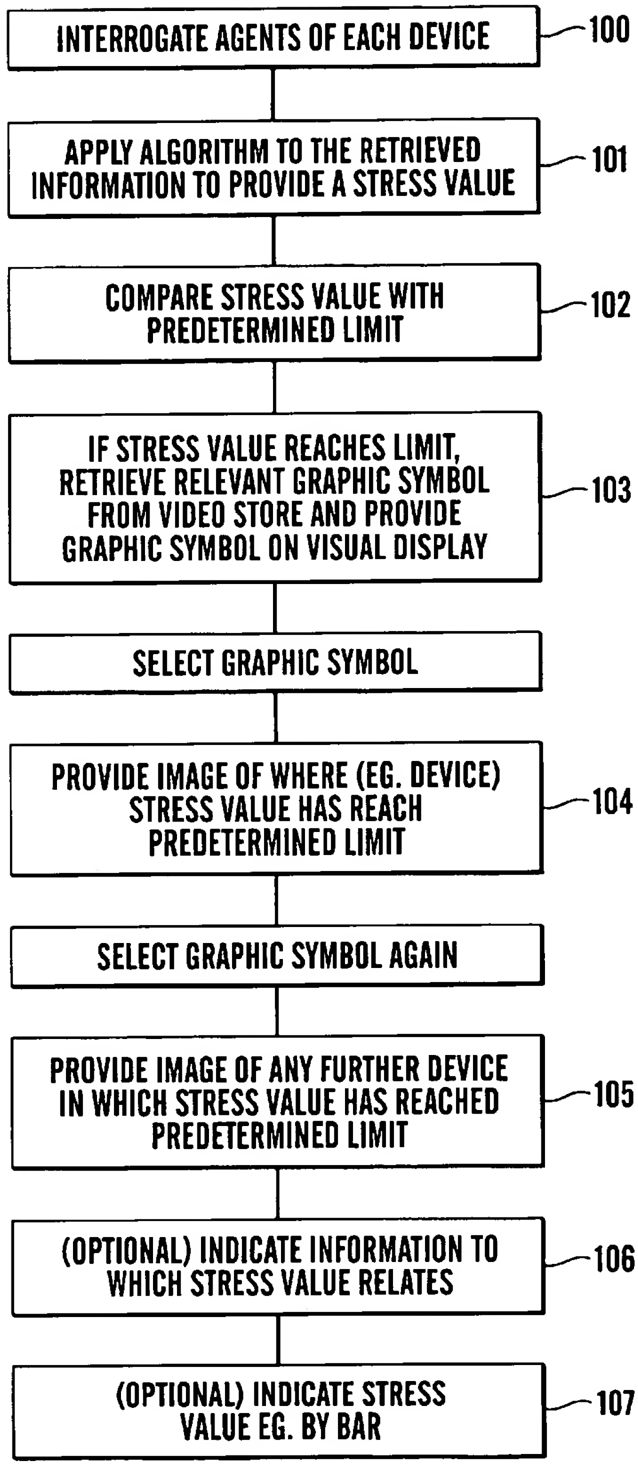

INTERROGATE AGENTS OF EACH DEVICE
100
APPLY ALGORITHM TO THE RETRIEVED INFORMATION TO PROVIDE A STRESS VALUE
101
COMPARE STRESS VALUE WITH PREDETERMINED LIMIT
102
IF STRESS VALUE REACHES LIMIT, RETRIEVE RELEVANT GRAPHIC SYMBOL FROM VIDEO STORE AND PROVIDE GRAPHIC SYMBOL ON VISUAL DISPLAY
103
SELECT GRAPHIC SYMBOL
PROVIDE IMAGE OF WHERE (EG. DEVICE) STRESS VALUE HAS REACH PREDETERMINED LIMIT
104
SELECT GRAPHIC SYMBOL AGAIN
PROVIDE IMAGE OF ANY FURTHER DEVICE IN WHICH STRESS VALUE HAS REACHED PREDETERMINED LIMIT
105
(OPTIONAL) INDICATE INFORMATION TO WHICH STRESS VALUE RELATES
106
(OPTIONAL) INDICATE STRESS VALUE EG. BY BAR
107

Fig.2

NETWORK PROBLEM INDICATION

BACKGROUND OF THE INVENTION

The present invention relates to supervising a network, that is a network of electronic devices comprising, for example, workstations, personal computers, servers, hubs, routers, bridges, switches, (hereinafter referred to as devices of the network), and links between these devices which may be in the form of physical cable or wireless links. The network may be a local area network (LAN), wide area network (WAN) or other types and may operate in accordance with any desired protocol.

After such a network has been installed, it is desirable for the person appointed network manager to be able to understand the technical operation of the network. In known network management systems, the manner in which the relevant data is retrieved from the managed devices, compiled and displayed has been problematic in several respects. Firstly, the data received from each of the managed devices is simply compiled and displayed as a list of data for the user to interpret. Secondly, the data does not provide information about unmanaged devices. Thirdly, information about a given network device, such as the type of device, location of the device on the network and operating speed of the device, may be contained in different sections of the compiled data. Consequently, conventional systems are cumbersome and difficult to use.

In co-pending UK patent applications numbers 9910838.3 and 9910837.5 (each in the name of the assignee of the present application) which are incorporated herein, we describe various arrangements for providing interrogation of the devices of the network to thereby produce on a network manager's workstation details of the network and its operation (preferably in the form of a network map which may be displayed on a visual display unit showing the devices and links between the devices). At its simplest, and where the device is a "managed" device, this information is usually provided by interrogation using a known protocol, such as the SNMP protocol, of the so-called 'agent' of each device which stores the device's unique MAC address, the type of device and the MAC addresses of the devices which are connected to the ports directly or indirectly.

In our UK patent application 9910838.3 in particular there is disclosed a system which monitors a plurality of stress values or so-called "stress metrics" for the managed devices on the network and provides an overall stress value for each device or alternatively an overall stress value for the network as a whole. (There may also be stress values or metrics for components, eg chassis blades or ports of the device.)

metric value

Particularly where the network is a smaller network than would warrant a full time network manager, it is desirable for the network supervisor to be able to use his workstation for other tasks than network management and whilst the network supervisor is engaged in these other tasks, it is desirable if an indication can be given to the network supervisor that certain events, such as, but not exclusively, problems have occurred on the network. By "problems" we mean matters to which the network supervisor would like to give his attention and may include trivial or serious problems or other events.

Typical problems which may affect the performance of the network include:

1. Slow operating of the speed of the network, and individual network devices, leading to slow movement of data traffic across the network, indicated by, for example, slow response time for a given network device;

2. High volumes of data traffic on the network due to, for example, over utilisation of the networks links, network devices, and the network as a whole; and
3. High error rates in the transmission of data packets across the network, indicated by, for example, the loss of data packets in a network device and errors in received data packets.

SUMMARY OF THE INVENTION

The present invention provides a network supervising apparatus comprising:

- arithmetic apparatus for applying an algorithm to information relating to the devices of the network to provide a stress value,
- a comparator to compare the stress value with a predetermined limit,
- a processing unit to retrieve graphic symbol related signals from a signal store and to provide the graphic symbol on the visual display apparatus when the stress value reaches the predetermined limit,
- a manual selector for selecting the graphic symbol and causing said visual display apparatus to provide an image indicating where the stress value has reached the predetermined limit.

The present invention may also provide a computer program on a computer readable medium or embodied in a carrier wave for use in supervising a network, said program comprising:

- program step or means for applying an algorithm to information relating to the devices of the network to provide a stress value,
- program step or means to compare the stress value with a predetermined limit,
- program step or means for retrieving graphic symbol signals from a signal store and for providing the graphic symbol on a visual display apparatus when the stress value reaches the predetermined limit,
- program step or means actuated by manually selecting the graphic symbol to cause said visual display apparatus to provide an image indicating where the stress value has reached the predetermined limit.

The present invention also provides a method of supervising a network comprising:

- applying an algorithm to information relating to the devices of the network to provide a stress value,
- comparing the stress value with a predetermined limit, retrieving graphic symbol signals from a signal store and providing the graphic symbol on a visual display apparatus when the stress value reaches the predetermined limit,
- manually selecting the graphic symbol and causing said visual display apparatus to provide an image indicating where the stress value has reached the predetermined limit.

BRIEF DESCRIPTION OF THE DRAWINGS

Preferred embodiments of the invention will now be described by way of example only with reference to the accompanying drawings in which:

FIG. 1 is a diagrammatic view of a network incorporating a preferred embodiment of the invention, and

FIG. 2 sets out the program steps in accordance with the preferred embodiment of the invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring to FIG. 1 there is shown a network 10 comprising a plurality of devices in the form of a network

supervisor's workstation or computer **11**, other workstations **12B–E**, hubs **13A**, **13B**, and switch **14**. The network is a simple network and is set out for purposes of illustration only. Other configurations and arrangements, may be used.

The devices are connected together by means of links **16A–H** which may be hard wired and utilise any desired protocol, and link **16F** which is a wireless link.

The network supervisor's workstation includes, in addition to a visual display unit **18**, a central processing unit or signal processor **19**, a selector which may be in the form of a mouse **22**, a program store **21** which may comprise, for example, a CD drive, a floppy disk drive or a zip drive, and a memory **17** for storing a program which may have been loaded from the program store **21** or downloaded for example via Internet from a website.

By means which is disclosed in the co-pending patent applications referred to above, the network supervisor's computer **11** may interrogate and analyse the network, and store in the memory **17** the information relating to the devices within the network and the links between the devices. In essence, most quality devices include a so-called agent which stores information about the device such as how many ports it has and how they are connected, and the address to which at least some of the ports are connected, and its unique MAC number, its Sys Object ID which identifies what the device is and what model type it is. The computer **11** interrogates the agents of each device.

topology

In a preferred arrangement, the computer **11** may, on command from the selector **22**, process signals from the memory **17** by the signal processor **19** and provide on the visual display unit **18** a network map showing each of the devices and the links therebetween. In the examples shown, the network is simple but of course in many instances the network will be considerably more complex and it may be necessary to arrange that the visual display unit **18** only shows a simplified version or only part of the network at any one time.

In addition to the above and as described in our co-pending UK patent application 9910838.3 the computer **11** includes arithmetic means to apply an algorithm to the information received by the interrogation of the agents of the devices to provide stress values for parts of each managed device (eg each port or each chassis blade) and/or for each managed devices on the network and/or for parts of the network and/or for the network as a whole. The computer **11** includes an arithmetic means to compare the or each stress value with a relevant predetermined threshold limit and provide a stress metric. The threshold limit may be selected by the network manager himself or may be pre set by the program.

Stress data

In other known network management arrangements, the network manager's computer is dedicated to network management and it is known to provide a visible indication of faults in the network. The prior systems are designed, however, to operate with large networks having one or more full time dedicated network managers. In the present case, whilst the network is operating, the network manager may carry out other tasks by running other unrelated programs on his computer **11**. However, it is desirable for him to be warned when the stress value for any individual device or for the network as a whole reaches or exceeds the threshold limit, even whilst he is carrying out completely unrelated tasks on his computer **11**.

The arrangement is such that whilst the network manager is using his workstation for other tasks, and in particular for tasks involving Windows (RTM) software, an indication should be provided on the screen to alert the network manager when the stress value of the network, part of the network (eg a branch), a device, or part of a device (eg a port or a chassis blade) reaches the predetermined limit. This provides substantial technical benefit in the operation of a network and is particularly useful when the network manager does not manage the network full time (eg where the network is small or relatively small). It will be seen therefore that the preferred arrangement allows the benefits of a large network with a full time manager to be applied to a smaller network where it is not sensible to employ a full time manager but allows a person with other duties to act part time as network manager and be warned of network problems whilst carrying out other computer based tasks, i.e. running other unrelated programs on his computer.

Indicator

Whilst such an indication can be provided in a variety of ways, it is preferred that the indication be in the form of a graphic symbol which appears on the network manager's screen, no matter what program he is running at the time. Preferably, it will appear in a predetermined set place on the screen so as to be obvious and recognisable. Particularly conveniently, the graphic symbol may appear in the so called "system tray" (which is provided in the form of a bar at the bottom of the Windows screen) or application status bar in the Windows screen. The Windows software includes a sub program which may be invoked to provide a graphic symbol in the system tray or application status bar.

Thus the computer **11** includes in its memory **17** (or elsewhere) signals relating to a relevant graphic symbol (which might be an icon or may be a text message such as "NETWORK PROBLEM") which may be chosen by the network vendor, and the software operated by the computer **11** includes a link to the necessary Windows software so that when the arithmetic apparatus calculates that the stress metric of a single device or network reaches or exceeds a predetermined limit, the signals relating to the graphic symbol are retrieved and a graphic symbol is displayed in the system tray of the Windows display.

The software is then arranged such that when the network manager uses the mouse to move a pointer to the relevant graphic symbol and clicks on the mouse, the existing Windows display is overlaid with a display from the network management software which provides details of the network problem.

This display may be simply in the form of a text message indicating the particular device, part of a device or part of the network which has reached the predetermined stress level, or may provide the network map or relevant part of the network map with the relevant device or devices highlighted, or may simply display an enlarged view of the relevant device.

Text

map highlight size

There will frequently be more than one device affected (eg when there is an overload) and successive clicks on the icon will similarly identify successively the relevant devices.

In this way the network manager can allow the network to run and only needs to deal with the network when a problem is indicated by means of the relevant graphic symbol in the system tray or application status bar of the Windows display.

In addition to providing an image of the relevant device, there may be provided an image of the "stress bar" referred to in our co-pending UK patent application 9910837.5. This will give the network manager a clearer indication of the level of stress in respect of that particular device.

stress bar

We have described how the network may be supervised. The preferred method of the invention is carried out under the control of the network supervisor's workstation or computer and in particular by means of a program controlling the processor apparatus of that computer or elsewhere in the system.

The program for controlling the operation of the invention may be provided on a computer readable medium, such as a CD, or a floppy disk, or a zip drive disk carrying the program or their equivalent, or may be provided on a computer or computer memory carrying the website of, for example, the supplier of the network products. The program may be downloaded from whichever appropriate source via, for example, a telephone line, a wireless radio or infra-red link, in each of which cases it may be embodied in a carrier wave and used to control the processor to carry out the steps of the invention as described.

The program may include,

a program step or means (**100**) for interrogating the agents of each device,

a program step or means (**101**) for applying an algorithm to the information relating to the devices interrogated to provide a stress value,

program step or means (**102**) to compare the stress value with a predetermined limit,

program step or means (**103**) for retrieving graphic symbol signals from a signal store and for providing the graphic symbol on the visual display apparatus when the stress value reaches the predetermined limit (even if the workstation is running another program and the visual display apparatus is displaying a visual display controlled by an unrelated program),

program means (**104**) actuated by the mouse or its equivalent selecting the graphic symbol and causing the visual display on the visual display apparatus to change to provide an image indicating where the stress value has reached the predetermined limit,

program means (**105**) actuated by successive selections of the graphic symbol by the mouse to display further devices in which the stress value has reached a predetermined limit,

(optionally) program means (**106**) to indicate the information to which the stress value relates, and

(optionally) program means (**107**) to indicate the value of the stress value (eg by means of the bar described in UK patent application 9910837.5)

Program step (**100**) may be provided by another program.

Thus the preferred arrangement of the invention allows the application to visually inform the network manager when network problems occur and allows the user to pinpoint the relevant devices causing these problems simply by clicking on the graphic symbol. This reduces the amount of time taken to resolve network problems by allowing the network manager to find out immediately which devices are at fault. By displaying the symbol in either the status bar or Windows system tray, the user does not constantly have to scan the network map to determine if errors have occurred. The use of the Windows system tray allows the user to keep track of network problems whilst using the workstation for other tasks.

The invention is not restricted to the details of the foregoing example.

What is claimed is:

1. A network supervising apparatus comprising:

a computer capable of running a plurality of programs, said computer including,

arithmetic apparatus (i) applying, whilst said computer is running a network supervising program, an algorithm to information relating to the devices of a network to provide a stress value; and (ii) comparing the stress value with a predetermined limit,

a processing unit retrieving graphic symbol related signals from a signal store and providing the graphic symbol on the visual display apparatus when the stress value reaches the predetermined limit, said processing unit being operable to provide said graphic symbol on the visual display apparatus when said computer is running an unrelated program, and

a selector selecting the graphic symbol and causing the visual image displayed on said visual display apparatus to change to provide an image indicating where the stress value has reached the predetermined limit.

2. A network supervising apparatus further comprising interrogation means interrogating agents of the devices of the network to provide the information for the arithmetic means.

3. A network supervising apparatus as claimed in claim **1** in which the graphic symbol comprises a text message.

4. A network supervising apparatus as claimed in claim **1** in which the graphic symbol comprises an icon.

5. A network supervising apparatus as claimed in claim **1** in which the graphic symbol is provided in a set predetermined position on the visual display on the visual display apparatus.

6. A network supervising apparatus as claimed in claim **1** in which the visual display apparatus displays a Windows based display and the graphic symbol is provided in the system tray.

7. A network supervising apparatus as claimed in claim **1** in which the visual display apparatus displays a Windows based display and the graphic symbol is provided in the application status bar.

8. A network supervising apparatus as claimed in claim **1** in which the stress value relates to a port of a device.

9. A network supervising apparatus as claimed in claim **1** in which the stress value relates to a chassis blade of a device.

10. A network supervising apparatus as claimed in claim **1** in which the stress value relates to a device.

11. A network supervising apparatus as claimed in claim **1** in which the stress value relates to part of the network.

12. A network supervising apparatus as claimed in claim **1** in which the stress value relates to the speed of operation of the relevant part of the network.

13. A network supervising apparatus as claimed in claim **1** in which the stress value relates to the error rate in the transmission of data packets of the relevant part of the network.

14. A network supervising apparatus as claimed in claim **1** in which the stress value relates to the volume of data traffic on the relevant part of the network.

15. A computer program on a computer readable medium for use in supervising a network, said program comprising:

program step for applying an algorithm to information relating to the devices of a network to provide a stress value,

program step to compare the stress value with a predetermined limit,

program step for retrieving graphic symbol signals from a signal store and for providing the graphic symbol on a visual display apparatus when the stress value reaches the predetermined limit and whilst said visual display

apparatus is displaying a visual display controlled by an unrelated program,

program step actuated by manually selecting the graphic symbol to cause said visual display on said visual display apparatus to change to provide an image indicating where the stress value has reached the predetermined limit.

16. A computer program as claimed in claim 15 including an initial program step to interrogate agents of the devices of the network to provide the information to which the algorithm is to be applied.

17. A computer program as claimed in claim 15 in which the graphic symbol comprises a text message.

18. A computer program as claimed in claim 15 in which the graphic symbol comprises an icon.

19. A computer program as claimed in claim 15 in which the visual display apparatus displays a Windows based display and the graphic symbol is provided in the system tray.

20. A network supervising apparatus as claimed in claim 1 in which the graphic symbol is provided in a set predetermined position on the visual display on the visual display apparatus.

21. A computer program as claimed in claim 15 in which the visual display apparatus displays a Windows based display and the graphic symbol is provided in the application status bar.

22. A computer program as claimed in claim 15 in which the stress value relates to a port of a device.

23. A computer program as claimed in claim 15 in which the stress value relates to a chassis blade of a device.

24. A computer program as claimed in claim 15 in which the stress value relates to a device.

25. A computer program as claimed in claim 15 in which the stress value relates to part of the network.

26. A computer program as claimed in claim 15 in which the stress value relates to the speed of operation of the relevant part of the network.

27. A computer program as claimed in claim 15 in which the stress value relates to the error rate in the transmission of data packets of the relevant part of the network.

28. A computer program as claimed in claims 15 in which the stress value relates to the volume of data traffic on the relevant part of the network.

29. A method of supervising a network comprising:

applying an algorithm to information relating to the devices of a network to provide a stress value,

comparing the stress value with a predetermined limit,

retrieving graphic symbol signals from a signal store and providing the graphic symbol on a visual display apparatus when the stress value reaches the predetermined limit and whilst said visual display apparatus is displaying a visual display controlled by an unrelated program,

manually selecting the graphic symbol and causing said visual display on said visual display apparatus to change to provide an image indicating where the stress value has reached the predetermined limit.

30. A method as claimed in claim 29 including an initial step of interrogating agents of the devices of the network to provide the information to which the algorithm is applied.

31. A method as claimed in claim 29 in which the graphic symbol comprises a text message.

32. A method as claimed in claim 29 in which the graphic symbol comprises an icon.

33. A method as claimed in claim 1 in which the graphic symbol is provided in a set predetermined position on the visual display on the visual display apparatus.

34. A method as claimed in claim 29 in which the visual display apparatus displays a Windows based display and the graphic symbol is provided in the system tray.

35. A method as claimed in claim 29 in which the visual display apparatus displays a Windows based display and the graphic symbol is provided in the application status bar.

36. A method as claimed in claim 29 in which the stress value relates to a port of a device.

37. A method as claimed in claim 29 in which the stress value relates to a chassis blade of a device.

38. A method as claimed in claim 29 in which the stress value relates to a device.

39. A method as claimed in claim 29 in which the stress value relates to part of the network.

40. A method as claimed in claim 29 in which the stress value relates to the speed of operation of the relevant part of the network.

41. A method as claimed in claim 29 in which the stress value relates to the error rate in the transmission of data packets of the relevant part of the network.

42. A method as claimed in claim 29 in which the stress value relates to the volume of data traffic on the relevant part of the network.

43. A computer program on a computer readable medium, said computer program comprising software for performing the method of claim 29.

44. A computer program embodied in a carrier wave for use in supervising a network, said program comprising:

program step for applying an algorithm to information relating to the devices of a network to provide a stress value,

program step to compare the stress value with a predetermined limit,

program step for retrieving graphic symbol signals from a signal store and for providing the graphic symbol on a visual display apparatus when the stress value reaches the predetermined limit and whilst said visual display apparatus is displaying a visual display controlled by an unrelated program,

program step actuated by manually selecting the graphic symbol to cause said visual display on said visual display apparatus to change to provide an image indicating where the stress value has reached the predetermined limit.

45. A computer program as claimed in claim 44 including an initial program step to interrogate agents of the devices of the network to provide the information to which the algorithm is to be applied.

46. A computer program as claimed in claim 44 in which the graphic symbol comprises a text message.

47. A computer program as claimed in claim 44 in which the graphic symbol comprises an icon.

48. A network supervising apparatus as claimed in claim 44 in which the graphic symbol is provided in a set predetermined position on the visual display on the visual display apparatus.

49. A computer program as claimed in claim 44 in which the visual display apparatus displays a Windows based display and the graphic symbol is provided in the system tray.

50. A computer program as claimed in claim 44 in which the visual display apparatus displays a Windows based display and the graphic symbol is provided in the application status bar.

51. A computer program as claimed in claim 44 in which the stress value relates to a port of a device.

52. A computer program as claimed in claim 44 in which the stress value relates to a chassis blade of a device.

53. A computer program as claimed in claim 44 in which the stress value relates to a device.

54. A computer program as claimed in claim 44 in which the stress value relates to part of the network.

55. A computer program as claimed in claim 44 in which the stress value relates to the speed of operation of the relevant part of the network.

56. A computer program as claimed in claim 44 in which the stress value relates to the error rate in the transmission of data packets of the relevant part of the network.

57. A computer program as claimed in claims 44 in which the stress value relates to the volume of data traffic on the relevant part of the network.

* * * * *

US005999604A

United States Patent [19]

Walter

[11] **Patent Number: 5,999,604**

[45] **Date of Patent: Dec. 7, 1999**

[54] **SYSTEM AND METHOD FOR MANAGING A TELECOMMUNICATIONS NETWORK BY DETERMINING SERVICE IMPACT**

[75] Inventor: **Craig Walter**, Colorado Springs, Colo.

[73] Assignee: **MCI Communications Corporation**, Washington, D.C.

[21] Appl. No.: **09/033,705**

[22] Filed: **Mar. 3, 1998**

[51] **Int. Cl.^{6}** **H04M 15/00**

[52] **U.S. Cl.** **379/133**; 379/134; 379/112; 379/113

[58] **Field of Search** 379/111–115, 120, 379/127, 126, 133–134, 137, 139–140, 116–119

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,606,601 2/1997 Witzman et al. 379/113

5,627,886 5/1997 Bowman 379/111

5,854,834 12/1998 Gottlieb et al. 379/113

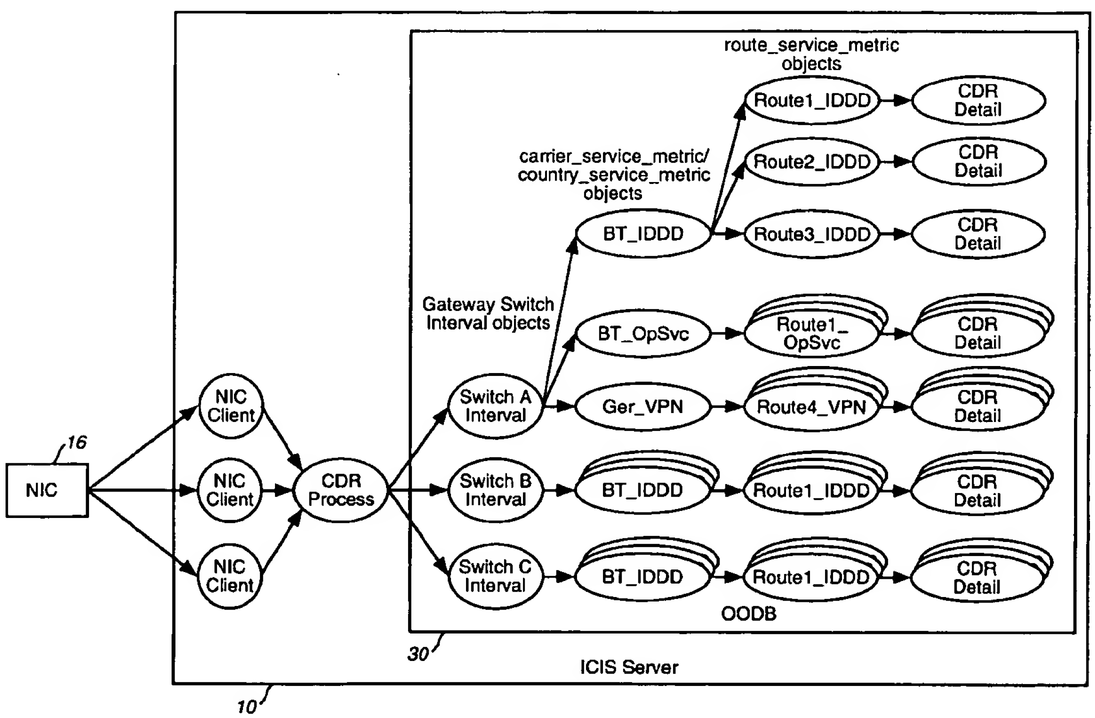
Primary Examiner—Huyen Le

Assistant Examiner—Duc Nguyen

[57] **ABSTRACT**

A system and method for managing a telecommunications network through near real time analysis of Call Detail Records (CDRs) is provided. The system and method of the invention reports how a particular call service is performing and how its performance deteriorates as a result of any type of network event, including both hardware and software related events. A feature of the invention is that it provides several levels of service impact reporting, thus providing users with more versatile and detailed network management capabilities.

5 Claims, 16 Drawing Sheets



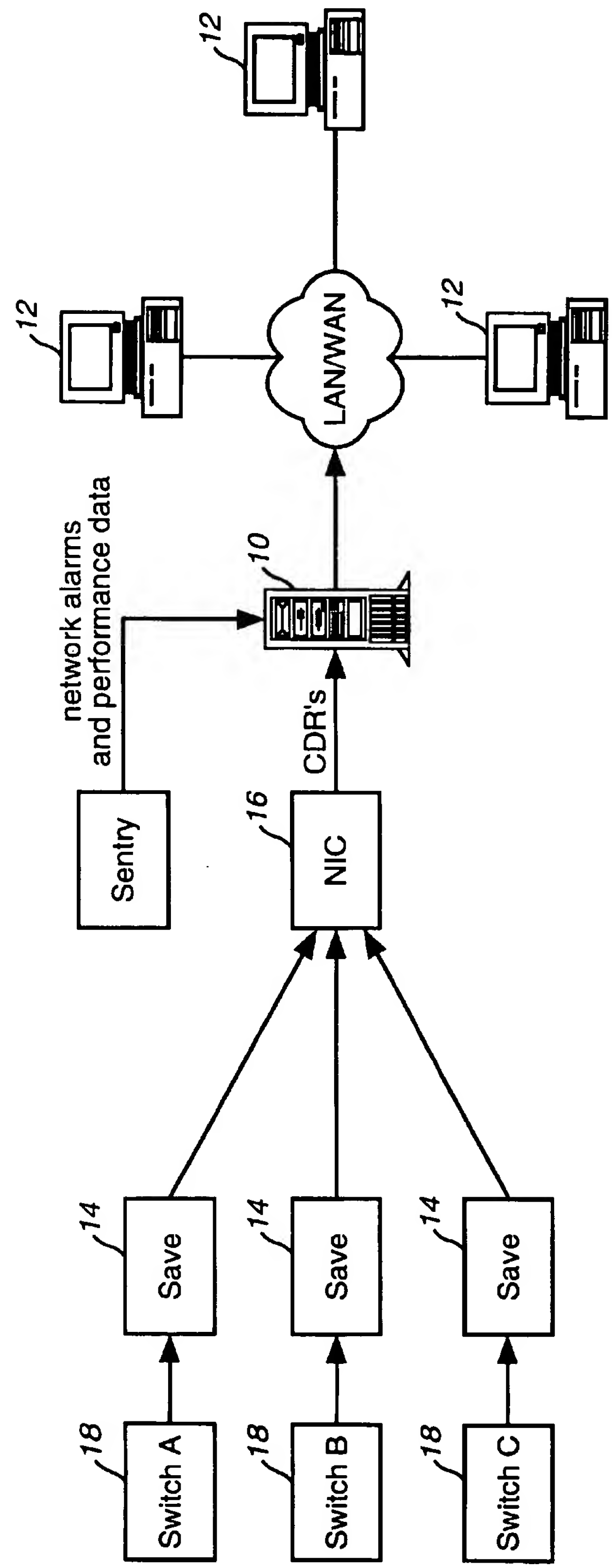
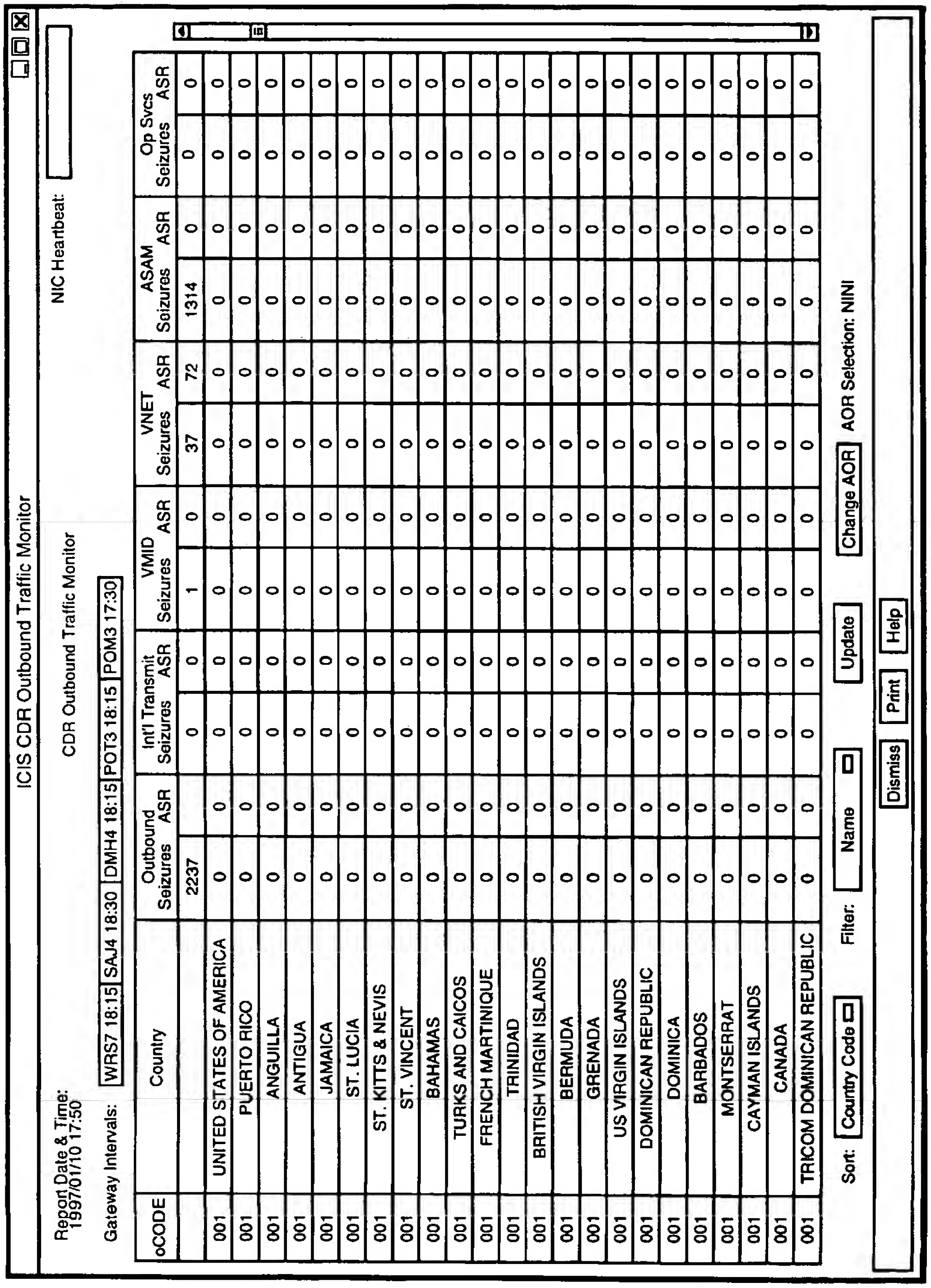

12
12
12
LAN/WAN
network alarms
and performance data
10
CDR's
Sentry
16
NIC
14
Save
14
Save
14
Save
18
Switch A
18
Switch B
18
Switch C

FIG. 1



oCODE	Country	Outbound Seizures	Outbound ASR	Int'l Transmit Seizures	Int'l Transmit ASR	VMID Seizures	VMID ASR	VNET Seizures	VNET ASR	ASAM Seizures	ASAM ASR	Op Svcs Seizures	Op Svcs ASR
		2237	0	0	0	1	0	37	72	1314	0	0	0
001	UNITED STATES OF AMERICA	0	0	0	0	0	0	0	0	0	0	0	0
001	PUERTO RICO	0	0	0	0	0	0	0	0	0	0	0	0
001	ANGUILLA	0	0	0	0	0	0	0	0	0	0	0	0
001	ANTIGUA	0	0	0	0	0	0	0	0	0	0	0	0
001	JAMAICA	0	0	0	0	0	0	0	0	0	0	0	0
001	ST. LUCIA	0	0	0	0	0	0	0	0	0	0	0	0
001	ST. KITTS & NEVIS	0	0	0	0	0	0	0	0	0	0	0	0
001	ST. VINCENT	0	0	0	0	0	0	0	0	0	0	0	0
001	BAHAMAS	0	0	0	0	0	0	0	0	0	0	0	0
001	TURKS AND CAICOS	0	0	0	0	0	0	0	0	0	0	0	0
001	FRENCH MARTINIQUE	0	0	0	0	0	0	0	0	0	0	0	0
001	TRINIDAD	0	0	0	0	0	0	0	0	0	0	0	0
001	BRITISH VIRGIN ISLANDS	0	0	0	0	0	0	0	0	0	0	0	0
001	BERMUDA	0	0	0	0	0	0	0	0	0	0	0	0
001	GRENADA	0	0	0	0	0	0	0	0	0	0	0	0
001	US VIRGIN ISLANDS	0	0	0	0	0	0	0	0	0	0	0	0
001	DOMINICAN REPUBLIC	0	0	0	0	0	0	0	0	0	0	0	0
001	DOMINICA	0	0	0	0	0	0	0	0	0	0	0	0
001	BARBADOS	0	0	0	0	0	0	0	0	0	0	0	0
001	MONTSERRAT	0	0	0	0	0	0	0	0	0	0	0	0
001	CAYMAN ISLANDS	0	0	0	0	0	0	0	0	0	0	0	0
001	CANADA	0	0	0	0	0	0	0	0	0	0	0	0
001	TRICOM DOMINICAN REPUBLIC	0	0	0	0	0	0	0	0	0	0	0	0

FIG. 2

CDR Traffic for SINGAPORE's Inbound Direct Service

Report Date & Time:
1997/01/10 18:17

NIC Heartbeat:

Gateway Intervals: WRS7 18:15 | SAJ4 18:30 | DMH4 18:15 | POT3 18:15 | POM3 17:30

Route	Seizures	Answers	ASR	MHT	Failed ATTS	Tech Fault	Address Incomp	Unalloc Number	Equip Congest	IntlNet Congest	NatlNet Congest
DMH4 0448	278	115	41	35991	3	0	3	0	0	0	0
DMH4 1249	8	2	25	35892	1	0	1	0	0	0	0
DMH4	296	117	40	35989	4	0	4	0	0	0	0
SAJ4 0474	140	56	40	232	2	0	0	2	0	0	0
SAJ4 1245	119	52	43	146	0	0	0	0	0	0	0
SAJ4 0468	46	24	52	172	0	0	0	0	0	0	0
SAJ4 1249	221	93	42	156	0	0	0	0	0	0	0
SAJ4	526	225	42	174	2	0	0	2	0	0	0

Dismiss | Print | Help

FIG. 3

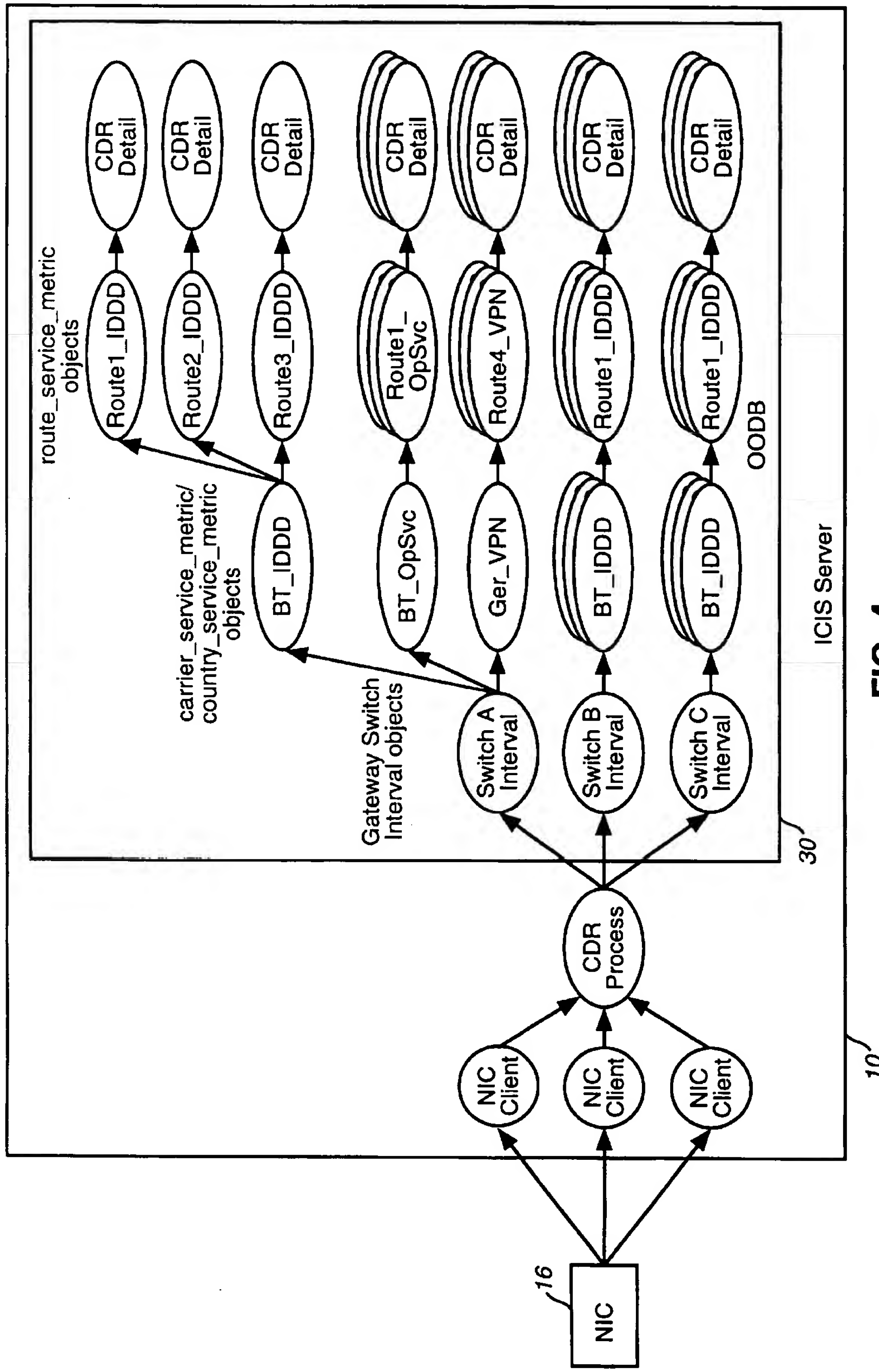

route_service_metric objects
carrier_service_metric/ country_service_metric objects
Gateway Switch Interval objects
CDR Detail
CDR Detail
CDR Detail
CDR Detail
CDR Detail
CDR Detail
CDR Detail
Route1_IDDD
Route2_IDDD
Route3_IDDD
Route1_ OpSvc
Route4_VPN
Route1_IDDD
Route1_IDDD
BT_IDDD
BT_OpSvc
Ger_VPN
BT_IDDD
BT_IDDD
Switch A Interval
Switch B Interval
Switch C Interval
OODB
30
ICIS Server
CDR Process
NIC Client
NIC Client
NIC Client
10
NIC
16

FIG. 4

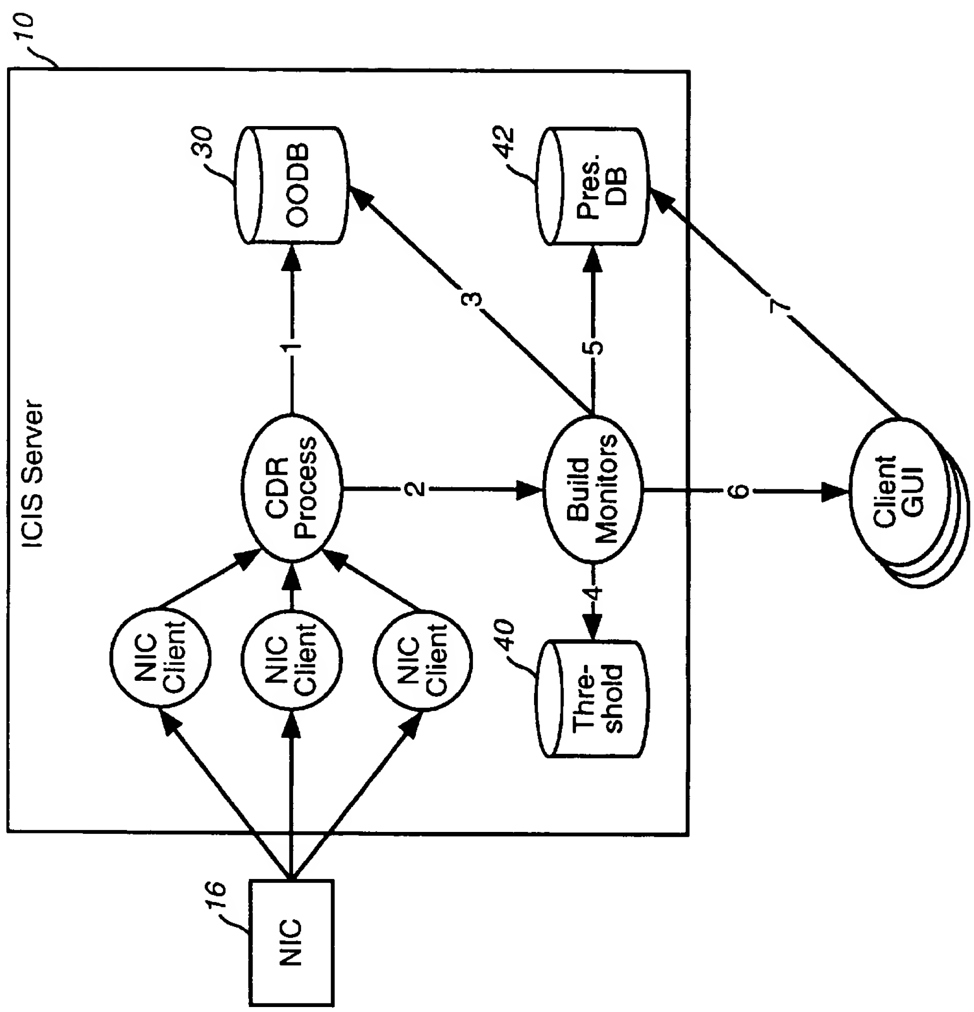

10
ICIS Server
30
OODB
42
Pres.
DB
CDR
Process
Build
Monitors
NIC
Client
NIC
Client
NIC
Client
40
Thre-
shold
Client
GUI
16
NIC
1
2
3
4
5
6
7

FIG. 5

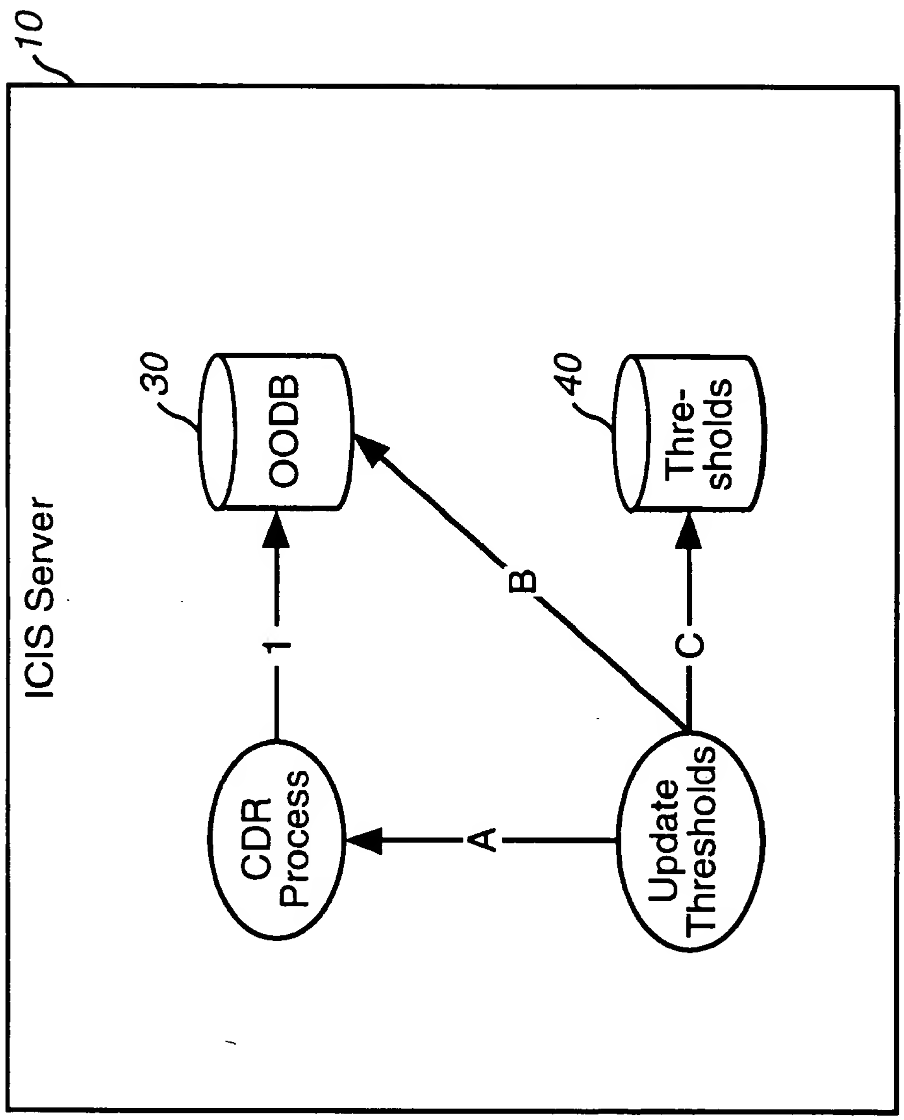

10
ICIS Server
CDR Process
1
30
OODB
B
40
Thre-sholds
C
A
Update Thresholds

FIG. 6

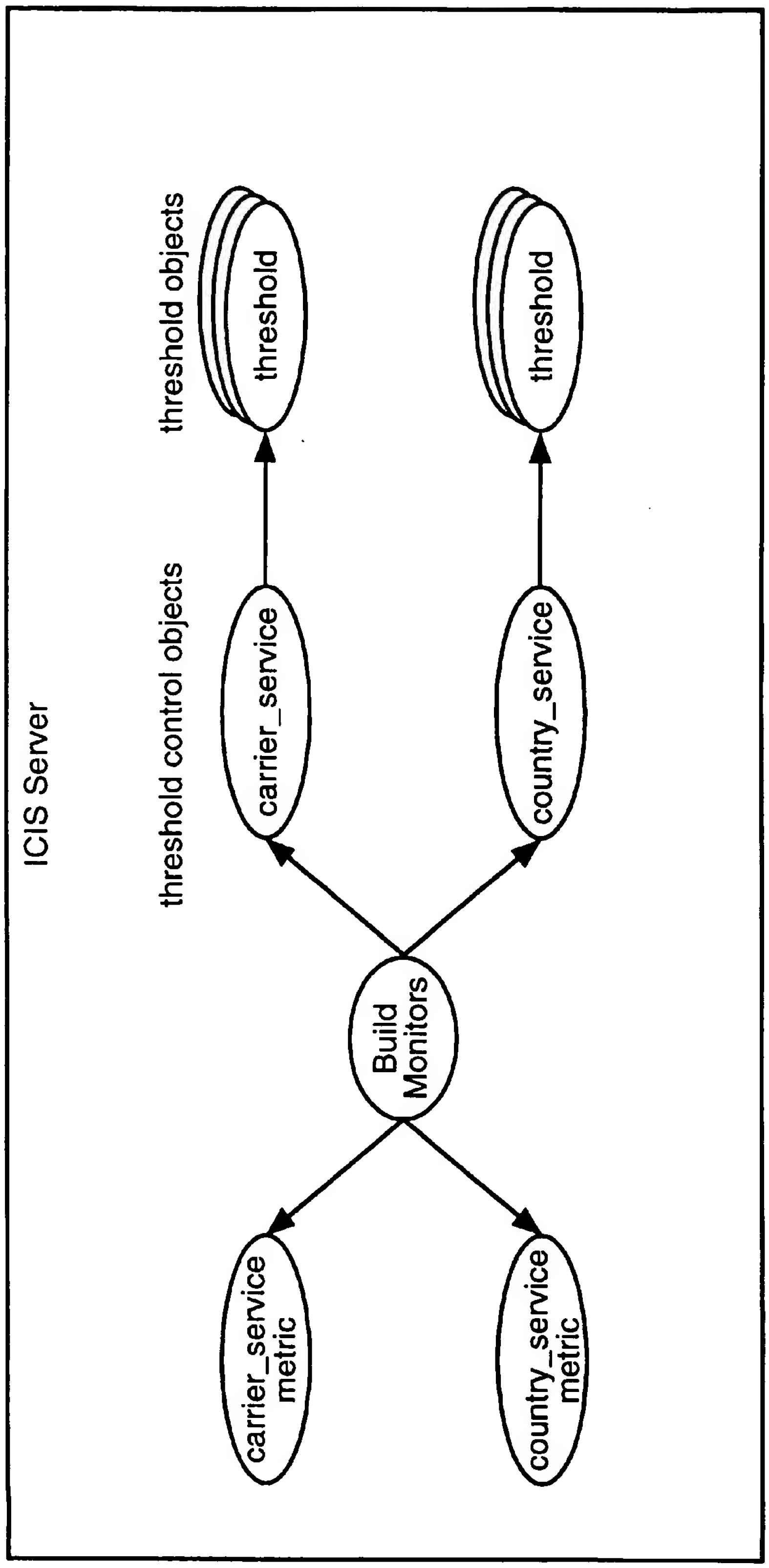

ICIS Server
threshold control objects
threshold objects
carrier_service
country_service
threshold
threshold
Build Monitors
carrier_service metric
country_service metric

FIG. 7

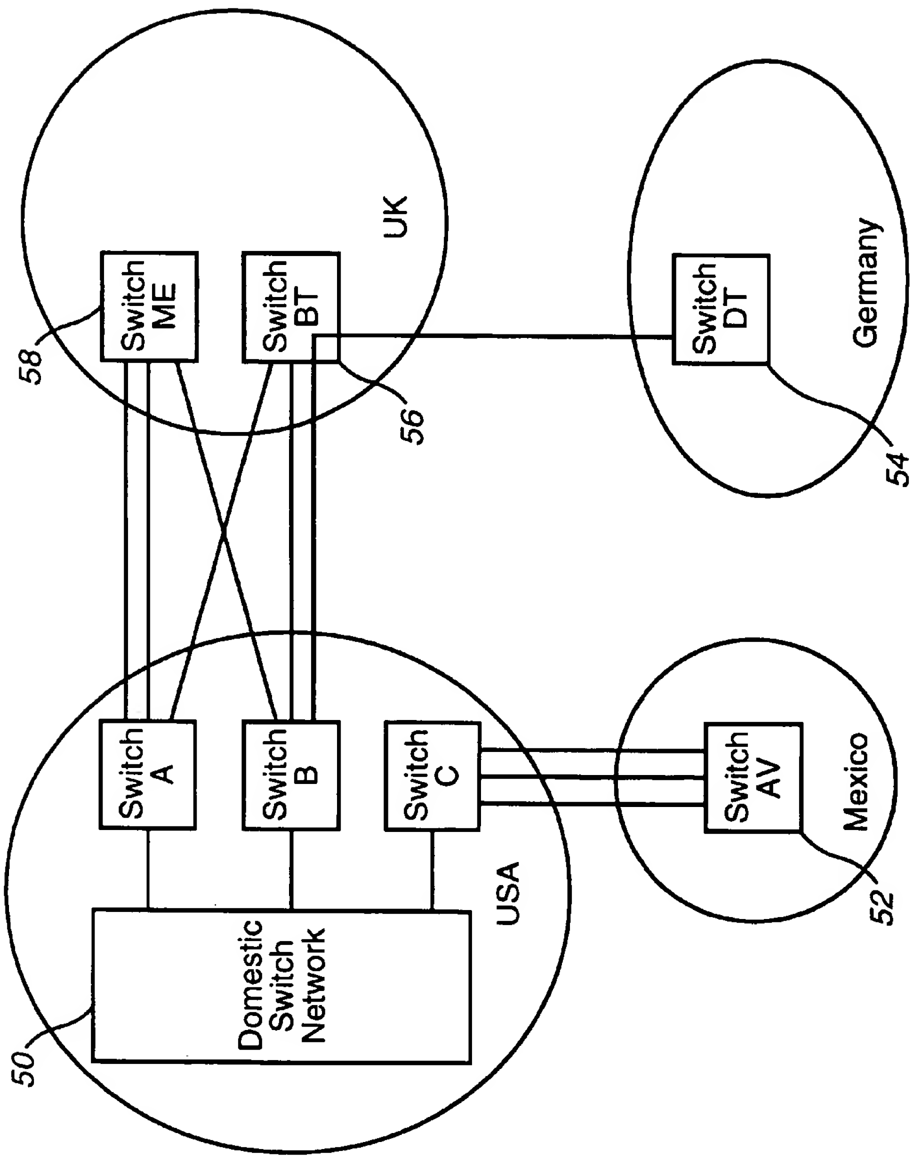

UK
Switch ME
Switch BT
58
56
Germany
Switch DT
54
USA
Switch A
Switch B
Switch C
Domestic Switch Network
50
Mexico
Switch AV
52

FIG. 8

ICIS CDR Detail

CDR Detail For Route POT3 0725, Inbound Direct Service

Dialed Digits	EOS Code	Count	EOS
Show	65535	415	Duplicate CDR
	3736	102	user busy OCAUse-4
	3160	69	Incomplete B-number received
	3731	12	Unequipped Circuit Identification Code hardware blocking after reception of final signal
	3745	7	no circuit/channel available OCAUse-14
	3737	6	no user responding OCAUse-5
	100	2	Vacant B number series .....
	3744	1	"normal, unspecified OCAUse-13"
619432	3736	42	user busy OCAUse-4
513221	3736	13	user busy OCAUse-4
312284	3731	12	Unequipped Circuit Identification Code hardware blocking after reception of final message
904234	3736	8	user busy OCAUse-4
801462	3160	7	Incomplete B-number received
903225	3160	7	Incomplete B-number received

Dismiss Print Help

FIG. 9

EOS	Cat	Description					
24		Called subscriber is busy.					
25	UN	Call to a vacant number after suppression of the interception category by a subscriber whose TCL category admits t					
27		Called subscribers line is out of order or blocked					
28		Called subscribers line is in line blocked					
30	TF	Forbidden traffic direction.					
33		Subscriber busy.					
34	NNC	No idle PBX lines.					
35	UN	Unobtainable number.					
36	NNC	All PBX lines are in line lockout.					
37	NNC	No PBX subroutes in given direction.					
38	NNC	Subscribers line out of service.					
43		Send special info tone.					
53	EC	Congestion in GSN or similar congestion state					
54	EC	Shortage of code sender devices.					
56	NNC	Congestion in national network.					
57	INC	Congestion in int'l network.					
60	UN	Unpermitted traffic case.					
70	EC	Congestion in GS after 3 reselections					
72		Hotline.					
73		Hotline after time out.					
76		Time releasing in incoming individual					
77	EC	Time release in RE.					
79	EC	Congestion in SS after possible reselection of subscriber line					
85	UN	Unpermitted B-digit received at traffic to own subscriber					
89	TF	Technical fault. EM blocking.					
90	INC	Congestion in all alternatives in the routing case					
91	UN	Analysis branch within a routing case not specified					
92	TF	Incorrect line signal on outgoing side					
94		Insufficient priority in correction with route analysis					
95	TF	Technical fault, EM blocking.					
98		Vacant A-number series.					
99	UN	Analysis error received if data is not specified for a traffic case switch, value, origin, TCL category or service not v					
100	UN	Vacant B number series.					
101		Code reserved for free use.					
102		Code reserved for free use.					
103		Code reserved for free use.					
104		Code reserved for free use.					
105		Code reserved for free use.					
106		Code reserved for free use.					
107		Code reserved for free use.					
108		Code reserved for free use.					
109		Code reserved for free use.					
110		Code reserved for free use.					
111		Code reserved for free use.					
112		Code reserved for free use.					
113		Code reserved for free use.					
114		Code reserved for free use.					
115		Code reserved for free use.					

FIG. 10A

116		Code reserved for free use.
117		Code reserved for free use.
118		Code reserved for free use.
119		messages in route analysis. Used to give, I.e. alternative messages or actions in connection w/congestion
140	TF	Seizure acknowledgement not received. (time release before seizure acknowledgement
141	EC	Time release during digit transmission
142	TF	Blocking from line at seizure or glare
144	TF	Technical fault on outgoing side in home exchange
146	TF	Path selection failed at seizure of CS device
150	TF	Defective register signal received on the outgoing side
151	TF	Unpermitted digit received from RE
154	TF	Wrong state. Reselection not possible
156		Free codes for RA.
157		Free codes for RA.
158		Free codes for RA.
159		Free codes for RA.
160	TF	Technical fault on incoming side. EM blocking or faulty MF signal received.
162	TF	Incorrect signal received in CR. Signal not defined in the signalling diagram.
163	TF	Incorrect line signal received on incoming side.
164	EC	Overflow in IT digit store before register is connected. (Timeout before first digit)
165	TF	Timeout between digits.
166	TF	Technical fault on incoming side or EM blocking.
167	TF	Technical fault on outgoing side or EM blocking.
168	EC	Time release during digit transmission
169		No call acknowledgement signal has been received from PABX
170		Code reserved for free use
171		Code reserved for free use
172		Code reserved for free use
173		Code reserved for free use
174		Code reserved for free use
175		Code reserved for free use
176		Code reserved for free use
177		Code reserved for free use
178		Code reserved for free use
179		Code reserved for free use
180		Code reserved for free use
181		Code reserved for free use
182		Code reserved for free use
183		Code reserved for free use
184		Code reserved for free use
185		Code reserved for free use
186	UN	Too many B number digits received.
187	EC	Time out for KP or ST signal.
188	EC	Time release during digit transmission.
189	TF	Technical fault on incoming side.
190	TF	Defective line signal or blocking from line received on outgoing side.
220		Received if a service is to be applied but which is handled by a superior exchange (service center)
221		Received if a service is to be applied but which is handled by a superior exchange (service center)
222		Received if a service is to be applied but which is handled by a superior exchange (service center)

FIG. 10B

223		Received if a service is to be applied but which is handled by a superior exchange (service center)					
239		Request for special info tone sending from AS					
321		Used by SUC, SUCAD, OCODE					
370		Illegal traffic case according to access barring					
371		Illegal traffic case according to access barring					
372		Illegal traffic case according to access barring					
373		Illegal traffic case according to access barring					
374		Illegal traffic case according to access barring					
375		Illegal traffic case according to access barring					
376		Illegal traffic case according to access barring					
377		Illegal traffic case according to access barring					
378		Illegal traffic case according to access barring					
379		Illegal traffic case according to access barring					
380		Illegal traffic case according to access barring					
381		Illegal traffic case according to access barring					
382		Illegal traffic case according to access barring					
383		Illegal traffic case according to access barring					
384		Illegal traffic case according to access barring					
385		Illegal traffic case according to access barring					
433	TF	Failure due to improper exchange data					
462		Access barred.					
467	TF	Digital path not provided.					
468	INC	Special congestion signal (CCS) received					
469		Reset message received after a backward call setup message					
500		Code reserved for free use in RA or DA at B number, route, time supervision, or EOS analysis					
520		Listed in the routing case for emergency routing					
521							
522		Listed at the end of a routing case for no additional route choices					
523	UN	Listed in B number analysis to prevent from dialing 0 after country code					
524							
525	AI	Listed in B number analysis to address incomplete					
526							
527							
528	UN	Listed in B number analysis to prevent from dialing 9 after the country code					
529							
530							
531							
533							
534							
570		Used to provide busy flash.					
575							
576							
580							
600							
601							
625	AI	Address incomplete.					
660	TF	TFC-Alarm received on incoming side					
661	TF	TFC-Alarm received on outgoing side					
675		Failed continuity check.					

FIG. 10C

676		Reset circuit signal received during digit transmission.
679	INC	Circuit group congestion (CGC) received during digit transmission
680	EC	Switching equipment congestion signal received.
681		Path selection failed at seizure of TC device
777		Repeat attempt.
810		No or incomplete A number received
811		No or incomplete A number received
858		Call failure signal received in C7.
859	TF	Timeout after last address signal.
860		Timeout waiting for continuity sig.
865	TF	Technical fault in code receiver and code sender
878		Destination blocking.
879		Subscriber not compatible.
902		Call is not to be allowed.
903		Call is to be allowed.
904		No available BWLI individual.
905		Function block is not active.
906	AI	Number collected from RE is incomplete.
907		Number of digits collected from RE is not between 6 and 10
1000		Terminating subscriber number in combination with test of out route
1114		Requested service not supported at Service Indicator Analysis
1115		B subscriber not compatible at terminating compatibility check
1193	1193	Outgoing internal translation failure die to improper exchange data
1608		B number info not defined in pre-analysis data
1609		A number info not defined in pre-analysis data
1610		Supplementary service info for the service User to User 'essential' is barred from sending due to routes
1611		Supplementary service info for the service closed to User 'w/o out access' is barred from sending due to incompatibl
1612		Supplementary service info for the service User to User 'essential' is barred from sending in a service screening cas
1614		Congestion in the CHS Kernel.
1622		Tariff message not convertible.
1655		Call collision (glare).
1656		Reception of unexpected message before first backward message on outgoing side
1657		Local HW blocking has occurred either on the incoming side or outgoing side after reception of the first backward r
1658		Local HW blocking has occurred either on the incoming side or outgoing side after reception of the first backward r
1659		Line HW blocking has occurred either on the incoming side or outgoing side after reception of the first backward m
1660		Line reset has occured on the outgoing side before reception of the first backwards message
1661		Line maintanence blocking has occurred on the outgoing side before the reception of the first backwards message
1662		Unsuccessful continuity check at call setup
1663	LIN	REL (1) - Unallocated number received
1664	INC	REL(3) - No route to destination received
1665		REL (4) - Send special tone.
1666		Release msg = Misdialed Trunk prefix.
1667		REL (16) - Normal clearing.
1668		REL (17) - User busy.
1669		REL (18) - No user responding.
1670		REL (19) - No answer from user.
1671		REL (21) - Call rejected.
1672	UN	REL (22) - Number changed.
1673		REL (27) - Destination out of order.

FIG. 10E

1674	UN	Invalid Number format- REL(28)					
1675		REL (29) - Facility rejected.					
1676		REL (31) - Normal unspecified.					
1677	NNC	REL(34) - No circuits available.					
1678		REL (38) - Network out of order.					
1679		REL (41) - Temporary Failure.					
1680	EC	REL (42) - Switching equipment congestion					
1681		REL (44) - Requested channel not available					
1682		REL (47) - Resource unavailable unspecified					
1683		REL (55) - Incoming calls barred withing CUG					
1684		REL (57) - Bearer capability not authorized					
1685		REL (58) - Bearer capability not presently available					
1686		REL (63) -Service or option not available unspecified					
1687		REL (65) - Bearer capability not implemented					
1688		Requested facility not implemented.					
1689		Only restricted digital info bearer capability is available					
1690		REL (79) - Service or option not implemented					
1691		REL (87) - User not a member of CUG					
1692		REL (88) - Incompatible destination					
1693		REL (95) - Invalid message unspecified					
1694		Message Type non-existent or not implemented					
1695		info element/parameter non-existent or not implemented					
1696		REL (102)- Recovery on timer expired.					
1697		Parameter non-existent or not implemented, passed on					
1698		REL (111) - Protocol error unspecified					
1699		REL (127) - Interworking unspecific					
2220		Requested service not supported at Telecommunications service analysis					
2324		Signalling network failure (outgoing)					
2402		Outgoing route not compatible at outgoing compatibility check					
2440	EC	Congestion in the charging output functions.					
2489		Request to link into traffic chain, refused.					
2746		User busy, no call back protection.					
2783		Supplementary service info for the service VPN is barred from sending in a service screening case					
2784		Supplementary service info for the service CUG 'w/o access' is barred from sending in a service screening case or n					
2785		Supplementary service info is not recognized for the service User to User essential by the succeeding exchange					
2786		Supplementary service info is not recognized for the service CUG w/o access by the succeeding exchange					
2793		Preemption.					
2794		Preemption, circuit reserved for reuse.					
2795		Subscriber absent.					
2796		Access info discarded.					
2797		Precedence call blocked.					
2798		Requested facility not subscribed.					
2799		Outgoing calls barred within CUG.					
2800		Inconsistency in designated outgoing...					
2801		Nonexistent CUG					
2802		Invalid transit network					
2803		Message with unrecognized parameter.					
3158	INC	No circuit available in transit network.					
3159		Reselection in the transit network applying					

FIG. 10F

3160	AI	Incomplete B-number received.					
3162		Overdecadic digit in conjunction with vacant number					
3167	UNN	More digits than the max number of Bdigit					
3168		Subscriber is not allowed to use the subscriber procedure					
3169	UNN	Incorrectly dialed subscriber procedure.					
3170		Carrier Analysis code not allowed.					
3171		Successful activation.					
3172		Interrogation, the presubscriber network operator services is activated					
3173		Interrogation, the presubscriber network operator services is not activated					
3174		Number of digits in CAC is faulty.					
3175		Successful deactivation.					
3176		Unsuccessful activation, dumping					
3177		Unsuccessful de-activation, dumping					
3178		CAC not defined in Equal Access analysis					
3179		CAC contains too many digits (overdecadic)					
3183		Allocated numbers.					
3190		User busy and diagnostic with CCBS is possible					
3407		No circuit available and diagnostic with CCBS possible.					
3408		No circuits available and diagnostics with CCBS possible					
3409		User busy and diagnostic with CCBS not possible					
3410		No circuits available and diagnostics with CCBS not possible					
3435		No data specified in the modification analysis					
3661	EC	Congestion in the audio controller.					
3670		OINF is not received from the incoming side					
3671		TINF is not received from the incoming side					
3672		Call type derivation not successful					
3673		Received dialling info not defined, dialled info between 0 and 99					
3674		Received dialling info not defined, dialled info between 100 and 127					
3675		Originating carrier analysis failed, because no originating carrier pointer is defined for incoming route					
3676		OCA failed, beOCAUse received multi carrier id is not defined for the carrier					
3732	UN	Unallocated number OCAUse=5					
3733		no route to destination OCAUse = 61					
3734		send special info tone OCAUse = 59					
3735		normal clearing OCAUse=3					
3736		user busy OCAUse=4					
3737		no user responding OCAUse=5					
3738		no answer from user OCAUse=61					
3739		Call rejected OCAUse=6					
3740		number changed OCAUse=7					
3741		destination out-of-order OCAUse=8					
3742	UN	invalid number format (address incompete) OCAUse=9					
3743		facility rejected (OCAUse=11)					
3744		normal, unspecified OCAUse=13					
3745		no circuit/channel available OCAUse=14					
3746		network out of order OCAUse=15					
3747		temporary failure OCAUse=16					
3748	EC	switching equipment congestion OCAUse=17					
3749		requested channel not available OCAUse=19					
3750		resource unavailable, unspecified OCAUse=20					

FIG. 10G

3751		incoming calls barred within CUG OCAUse=52
3752		Bearer capability not authorized OCAUse=21
3753		Bearer capability not presently available OCAUse=22
3754		Service or option not available or unspecified OCAUse=23
3755		Bearer capability not implemented. OCAUse=24
3756		Requested facility not implemented. OCAUse=53
3757		Only restricted digital info bearer capability is available OCAUse=26
3758		Service or option not implemented. OCAUse=27
3759		User not member of CUG. OCAUse=62
3760		Incompatible destination. OCAUse=34
3761		invalid or unspecified message. OCAUse=35
3762		Message type nonexistent or not implemented OCAUse = 37
3763		Parameter nonexisting or not implemented, discarded. OCAUse = 40
3764		Recovery on timer expiry. OCAUse=44
3765		Parameter nonexistent or not implemented, passed on. OCAUse = 43
3766		Protocol error, unspecified. OCAUse=45
3767		Interworking, unspecified. OCAUse=46
3768		Preemption OCAUse = 83
3769		Preemption circuit reserved for reuse. OCAUse=84
3770		Subscriber absent OCAUse=85
3771		Access information discarded. OCAUse = 18
3772		Precedence call blocked. OCAUse = 86
3773		Requested facility not subscribed. OCAUse=48
3774		Outgoing calls barred within CUG. OCAUse = 50
3775		Inconsistency in designated outgoing access info and SCL. OCAUse=87
3776		Nonexistent CUG. OCAUse=54
3777		Invalid transit network selection. OCAUse=88
3778		Message with unrecognized parameter, discarded. OCAUse=89
3779		No circuit available in transit network. OCAUse=103
3780		Reselection in the transit network applying load sharing. OCAUse=104
3781		Misdialed trunk prefix. OCAUse=60
3855	EC	No available echo control device.
3856		ISDN test call procedure failed. Specific route/device can not be selected, does not exist for the current traffic case
3859		Incompatible incoming signaling system.

FIG. 10H

SYSTEM AND METHOD FOR MANAGING A TELECOMMUNICATIONS NETWORK BY DETERMINING SERVICE IMPACT

FIELD OF THE INVENTION

This invention relates to telephone networks. More particularly the present invention relates to a network management system and method for analyzing Call Detail Records in near real time to detect, assess and report negative impacts to various call services.

BACKGROUND OF THE INVENTION

Conventional network management systems are comprised of network elements such as switches and transmission equipment. Network elements produce network event records such as report alarms responsive to the occurrence of network events. The term "network events", as used herein, refers to any event that may cause failure of network elements such as faulty transmission equipment, spliced cabling, switch abnormalities and the like. Network management systems are used to monitor communications networks and report alarms and other network events to users or system maintenance personnel.

Recent developments in network management systems enable networks to correlate many different network event records, for the purpose of showing users the correspondence between individual network events and individual network event records. Thus, users can track the effects of a given network event by analyzing the network event records produced responsive to the given network event. This is valuable because a single transmission outage, for example, can result in thousands of network event records being generated.

The present assignee has developed an international network management system, referred to as the International Community Information System (ICIS), that receives numerous network event records from switches and transmission network elements. Typical network event records include trunk alarms that report on the performance of a switch trunk. These network records are not related to any particular call or service. ICIS identifies and correlates switch alarms with transmission alarms, and presents information to the user indicative of the switch event records and transmission network event records that are related to a given network event. This method of network management identifies the network event records that are generated as a result of a given network event.

Typical communications networks support a variety of telephone services. Exemplary services include Inbound, Outbound and Outbound Transit. A limitation of prior art methods of network management is that they do not provide the user with an assessment of how telephone services are impacted by network events, including which services are impacted. This determination must be performed with manual analysis. In fact, many prior art systems are incapable of assessing service impact as a result of network events. This is because when a particular route fails, perhaps due to a fiber cut, calls are automatically switched to another route. This call switching is in accordance with normal network operations, and will go largely undetected as a problem. However, a particular service could be impacted by such an operation.

In addition, calls for a particular service can traverse several different routes. If there is a problem with a particular service, its impact would also traverse several different routes. Since prior art systems rely on the correlation of

reported network events, said correlation based on network topology, they would not be able to correlate the different instances of a service problem, since service problems are frequently not tied to network topology.

Another limitation of prior art methods and systems of network management is that they only detect and report problems that result from switching and transmission hardware failures. Often, a minor software fault, such as an error in a call routing translation database, can result in many blocked calls. Since such a problem would not cause the generation of network equipment alarms, it would go undetected by prior art network management systems.

SUMMARY OF THE INVENTION

The present invention is directed to a method and apparatus for analyzing service impacts on telecommunications networks through the analysis of Call Detail Records and call failure ratios. The invention utilizes Call Detail Records as a source of traffic data. Each call generates its own Call Detail Record. These records comprise a wealth of information relating to the calls including information regarding network events (call failures). In accordance with an aspect of the invention, multiple Call Detail Records are grouped or summarized according to various system parameters. For example, all Call Detail Records originating from the same source may be grouped together and Call Detail Records generated within a given time frame may be grouped together to form a multiplicity of subgroups.

time

Each of the Call Detail Records contained in the multiplicity of subgroups may be further grouped according to the service that carried the corresponding call that generated the Call Detail Record and according to the destination of the corresponding call.

From the forgoing subgroupings of Call Detail Records, the invention generates statistics indicative of a state of call services. Thus, a user is easily able to determine the impact of network events on call services.

In accordance with a first aspect of the invention, a method is provided for analyzing the effects of network events on call services. A plurality of call detail records (CDRs) are received from a plurality of network devices. Each of the CDRs includes a time stamp, call service data and call destination data. The plurality of CDRs are grouped into a plurality of first subgroups according to corresponding network devices and to the time at which they were produced. For each of the plurality of first subgroups, the CDRs are grouped into a plurality of second subgroups according to a call service data and call destination data. Call statistics are then generated indicative of a state of call services from data contained in the CDRs for each of the plurality of second subgroups. Thus, a user is readily able to identify which call services have been affected due to network events.

time

In accordance with another aspect of the invention, a programmed computer is provided for analyzing the effect of network events on call services. The computer is programmed to carry out a series of steps. The computer receives a plurality of call detail records (CDRs) from a plurality of network devices, each of the plurality of CDRs including a time stamp, call service data and call destination data. The plurality of CDRs are then grouped into a plurality of first subgroups according to corresponding network devices and to the time stamps of the plurality of CDRs. The computer rearranges the CDRs by subgroup. That is, for each of the plurality of first subgroups, the CDRs are grouped into a plurality of second subgroups according to a

call service data and call destination data. Call statistics are then generated indicative of a state of call services from data contained in the CDRs for each of the plurality of second subgroups. These statistics may be transmitted to remote terminals for display or they may be displayed by the programmed computer.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is an overall view of the system of the present invention.

FIG. 2 illustrates a screen print of a system display. The screen print includes information regarding outgoing calls.

FIG. 3 depicts another screen print of a system display. This screen print includes traffic information for Inbound Direct Calls.

FIG. 4 depicts the process architecture of an embodiment of the invention.

FIG. 5 shows internal process architecture for generating displays.

FIG. 6 illustrates further process architecture for generating thresholds.

FIG. 7 shows process architecture for applying thresholds.

FIG. 8 depicts a hardware diagram of an international telephone switching network.

FIG. 9 shows another screen print of a system display. The screen print includes route specific call information.

FIG. 10A–10H illustrate a list of call failure reasons.

DETAILED DESCRIPTION OF THE DRAWINGS

General Description

The invention is directed to a system and method for managing a telecommunications network, which may be comprised of various switches and transmission equipment. FIG. 1 illustrates network architecture in accordance with an embodiment of the invention. The system may be implemented as a software-based system that runs on a general purpose computer. In a preferred embodiment, the system may be realized through a client/server architecture, as shown in FIG. 1. In such a construction, a server 10 may embody software for data processing and analysis, and client computers (hereinafter clients) 12 may provide a Graphical User Interface (GUI) for graphic presentation of call metrics.

Unlike conventional systems, the present invention collects Call Detail Records (CDRs) from the switches 18 and extracts call metrics from the CDRs. Each CDR may include information regarding a particular call. More particularly a CDR may include a series of reserved fields of different lengths in which certain information may be inserted. Such fields may include, for example, END OF SELECTION CODE, TARIFF CLASS, DESTINATION ADDRESS (DIALED DIGITS) ORIGINATING TRUNK GROUP (ROUTE), DESTINATION TRUNK GROUP (ROUTE), GATEWAY ID (SWITCH ID), DISCONNECT TIME and HOLDING TIME as well as other fields. In addition, each CDR includes a timestamp to indicate when it was produced and when the call terminated. Typically, one CDR is created for the incoming port on which the call arrives, and another CDR is created for the outgoing port to which the call is routed. Thus, a single call which traverses one or more switches 18 may generate multiple CDRs. A computer system referred to as a Storage and Verification Element (SAVE) 14 collects, stores, and forwards all CDRs collected from a switch. The CDRs are forwarded to a Network Information Concentrator (NIC) 16, which collects all records generated by call processing elements (switches, service control points, etc.), formats records, and correlates records that are generated for the same call. The NIC 16 then provides processed call records to the server 10 for further processing. These CDRs are preferably provided within seconds of call completion. The clients 12 may then retrieve call metric information from the server 10 for graphical display. A preferred SAVE 14 is described and claimed in U.S. Pat. No. 5,606,601, assigned to the present assignee, and which is herein incorporated by reference. A preferred NIC 16 is described and claimed in U.S. pending patent application Ser. No. 08/426,256, assigned to the present assignee, and which is herein incorporated by reference.

In accordance with an aspect of the present invention, CDRs having like characteristics may be grouped together. More particularly, CDRs may be grouped according to their timestamps and in accordance with the switch from which they were received. Preferably, CDRs generated during a relatively short time interval are grouped together. This time interval is configurable and may be set to any duration, but in particularly preferred embodiments, it is set at about 15 minutes at the request of the user community.

In keeping with the invention, CDRs for a particular switch and time interval are then grouped in accordance with call service and destination metrics. The skilled artisan will readily recognize that there may be various categories of services depending upon the specific implementation of the invention. However, by way of example, in a preferred embodiment there are three categories of service, Outbound service, Inbound service and Outbound Transit service. Outbound service includes calls originating in the U.S. and terminating in a foreign country. Inbound service includes calls originating in a foreign country and terminating in the U.S. Outbound transit service includes calls originating in the U.S., destined for a first foreign country, and transferred to the first foreign country by a transit carrier in a second foreign country. A fourth category of service is Inbound transit service. This service is treated like Inbound service for the purposes of this invention. For each category of service, there are many individual services. A service for a call may be identified by a service code, which is included in the CDR. A service may also be identified by the destination address or "destination trunk group".

The call destination may be a country, a carrier, or both. In accordance with an aspect of the invention, each CDR may be placed in two destination/service groups: one in which the destination is a country and one in which the destination is a carrier. However, in a preferred embodiment, calls for inbound services are grouped by the carrier from which the call is received. The "originating trunk group" field in the CDR is preferably used to associate the call to a carrier. This mapping may be contained in a network topology database (not shown) that is readily accessible to the server 10. Calls for outbound services are preferably grouped by country. The country may be determined by the country code in the dialed digits, which may be included in the CDR. The mapping of country codes to countries may also be contained in the network topology database. Calls for outbound transit services are preferably grouped by the transit carrier. The "destination trunk group" field in the CDR may be used to associate the call to a carrier, the mapping for which may be contained in the network topology database. The final destination of an outbound transit service call may be determined by the country code in the dialed digits, and is therefore presented along with the transit carrier. An exemplary grouping is set forth in Table 1.

TABLE 1

Service	Route	Terminating Carrier	Dialed Country
Outbound Transit	WRG7 1023	British Telecom	40 (Germany)

In many cases, there is only one carrier for a country, and so it is sufficient to report only the country.

In furtherance of the invention, the CDRs in each group may be analyzed and statistical results may be generated and preferably displayed on the clients **12**. In a preferred embodiment, three different user presentation GUIs are presented. One GUI is presented for outbound services, one for inbound services, and one for outbound transit services. Inbound transit services are treated as inbound services. In the preferred embodiment, the outbound services GUI presents statistics by service and destination country, since a country can always be determined from the dialed number. The inbound services GUI presents statistics by service and carrier, since a carrier can always be determined by querying a network topology database with the trunk number on which the call was received. The transit services GUI presents statistics by service, final call destination (dialed country), and transit carrier. Each of the GUIs present statistics and call metrics in an readily accessible fashion. As an additional feature of the invention, the statistics and metrics may be color coded to indicate criticality.

The aforementioned presentations show the call statistics calculated from the analysis of CDRs collected over short time intervals. From these statistics, the user may easily detect a negative impact to a call service, regardless of whether the cause of the negative impact is know.

CDR ANALYSIS

In accordance with more detailed process aspects of the invention, CDRs may be analyzed to calculate many different statistics. In a preferred embodiment, at least two key statistics are calculated and presented. These are seizures and the Answer/Seizure Ratio (ASR). Seizures occur when the line between a gateway switch and a foreign carrier's switch is seized to setup a call. Seizures are sometimes referred to as call attempts. Seizures may be calculated by tallying the CDRs and excluding redundant CDRs (multiple CDRs are created for a single call). If a call fails for any reason, an indicator on the CDR designates a failure reason. Seizures minus failed calls equal answered calls. The ratio of answered calls to seizures is the ASR.

After a CDR is placed in a service/destination group, it may be counted as a seizure, and then either as a failed call attempt or an answered call. A call that encounters a ring-no-answer condition, for example, is considered a failed call, since it was not answered and is not billable. From the calculated number of seizures and answered calls, an ASR is calculated. The ASR is significant, as it provides the rate of answered, billable calls placed. Preferably, seizures and ASRs are calculated and presented in short time intervals, e.g. 15-minute intervals.

A failure reason may be provided on each CDR for failed calls. There are about 2000 failure reasons, which may be categorized into a number of failure categories. These categories were defined by the user community, and are discretionary. A list of EOS/Failure descriptions is illustrated in FIGS. **10A–10H**.

In addition to service and destination metrics, the present invention calculates statistics per route number and failure

category. When CDRs are grouped into service/destination groups, the data contained in each CDR is maintained, so that these lower-level calculations may be performed.

Referring to FIG. **2**, an outbound services screen is shown. The seizures and ASR for each service are presented by destination country, for a 15-minute interval. From this screen, additional details may be selected by the user. For example, when a user selects an ASR cell, a route-by-route breakdown of statistics for the selected service for the selected destination country is presented. This screen is shown in FIG. **3**. Preferably, the seizures and ASRs are presented for each route that supported the service during the interval. The route column includes both the gateway switch (i.e., DMH4) and the route number (i.e., 0448). Other statistics that are calculated from CDR data may also be shown, since these data are preserved with the CDR. A mean holding time (MHT) is calculated from the call duration data on CDRs, and indicates the mean time (averaged among all calls in a sample) the call was held on the network. The number of answered calls and failed calls, which may represent peg counts of qualified CDRs, also may be presented. From the identification, categorizing, and counting of failure codes on CDRs, the numbers of failed calls per failure category are presented.

From the route level detail screen, such as that shown in FIG. **3**, a user may obtain CDR detail presentation for a particular route, by selecting a route cell. A CDR detail screen for an inbound service route is shown in FIG. **9**. This screen shows call statistics for a single route, which extends from one particular gateway switch (i.e., POT3) to one particular switch of a foreign carrier, for a single service (Inbound Direct). Metrics presented include an End of Selection (EOS) code and CDR count (which equates to call count) for each dialed number prefix. The EOS code may be obtained directly from the CDR. It is preferably written to the CDR by the switch, and indicates what happened to the call, such as whether the call was successfully answered, or failed due to some reason. There is an EOS code for every call. For failed calls, the system uses the EOS code to categorize the call into one of the six failure categories. A dialed number prefix corresponds to a particular geographical calling area that is typically a small section of a city, perhaps covering a few square miles. Since this particular screen example is for an inbound service, dialed number prefixes are given as North American Numbering Plan (NANP) NPA-NXX numbers.

Using the CDR analysis method provided by the present invention, a user may detect an unusually low ASR for a service and destination country. However, different from prior art network management systems, the user may then "drill down" to the route level detail and identify a particular route that is contributing to the low ASR. From this level, a certain failure category may be seen as the reason for the low ASR. The user may then select that route and "drill down" to a CDR level detail and identify any particular calling areas for which a high number of failed calls are occurring. This level of detailed assessment of service impact, presented to users in a meaningful way, is a distinct advantage of the present invention.

An additional feature of the present invention is that each call metric on a presentation screen may be color coded to indicate a level of criticality of that metric. In the preferred embodiment, there are three levels: white indicates a first level, in which the metric falls within a threshold that is considered normal; yellow indicates a second level, in which the metric exceeds a first threshold but falls within a second threshold and is considered a warning; red indicates a third

level, in which the metric exceeds the second threshold and is considered critical. The color coding system is particularly advantageous as it quickly alerts users to metrics that are likely to cause problems.

Internal Process Architecture

The CDR analysis process described above may be performed by an appropriately programmed general purpose computer such as server 10. In a preferred embodiment, server 10 may be a mid-range computer such as a DEC Alpha server or an IBM Reporting System/6000. The above described CDR analysis process may be implemented in a variety of ways using various process architectures all of which are encompassed by the invention. FIG. 4 illustrates an exemplary internal process architecture of the server 10. In a preferred embodiment, the server 10 may include call metrics and processes in the form of instantiated objects. The server 10 may also embody an object-oriented database (OODB) 30 for storing collected and calculated data. Preferably, the instantiated object processes (with the exception of the NIC Clients and CDR Process) reside in the OODB 30, and each process writes the data that it calculates to the OODB 30.

metric

As shown by the process architecture of FIG. 4, CDRs are transmitted from the NIC 16 to the server 10. A plurality of NIC Client processes are used to manage communications with the NIC 16; they are shown for exemplary purposes in FIG. 4 and are further described in U.S. patent application Ser. No. 08/426,256. The NIC Clients may receive the CDRs in typical data record form, preferably TCP/IP packets, for example; extract the CDRs as data records; and forward the CDRs to CDR Process object. The CDR Process object groups CDRs into short time intervals, the time intervals being defined by the timestamp on the CDR, and not on the current system time of the server 10. However, CDRs are received from multiple gateway switches in different time intervals; that is, at a specific time, server 10 may receive a CDR from a first switch that is timestamped at 12:00:00 and a CDR from a second switch that is timestamped at 12:03:00. This is due to the different CDR processing loads on each switch. Therefore, CDR Process defines a short time interval for each switch, e.g. a 15-minute interval. These intervals are referred to as Switch Intervals. A Switch Interval is manifest as an object that is instantiated for a specific 15-minute interval for a specific gateway switch. As shown in FIG. 4, there is a Switch Interval object for each Switch (A, B, and C) in the network in FIG. 1.

When CDR Process object receives a CDR, CDR Process object first identifies the switch that created the CDR. If a Switch Interval object for that switch is currently open, CDR Process writes the CDR to that Switch Interval. Otherwise, CDR Process instantiates a new Switch Interval object for that switch. At this point, the Switch Interval object is manifest as temporary data storage in the server 10's memory, accessed only by CDR Process. As CDR Process object receives additional CDRs from that switch, it writes these CDRs to the open Switch Interval Object. In a preferred embodiment, when a CDR from a switch is received that is timestamped with a time equal to the short time interval, e.g., 15 minutes from the Switch Interval start time, the Switch Interval is completed, and CDR Process commits the Switch Interval object to the OODB 30.

Since Switch Intervals are objects, they may contain data (CDRs) and methods for processing that data. When CDR Process commits a Switch Interval object to the OODB 30, it triggers a method in that object for processing the CDRs in that object. This processing groups the CDRs by service and destination. Service may be explicitly given on each CDR by a service code; destination may be derived from the dialed number or route number or both, which may be given in the CDR. Services may be grouped into Inbound Services, Outbound Services, or Outbound Transit Services. Examples of Inbound Services include Direct (IDDD), International Transit, World Phone (an MCI service), Free Phone (an MCI service), toll-free, VPN (such as MCI's VNET), and Operator Services. Examples of Outbound Services include Direct, International Transit, World Wide Direct Dial, VPN, Automated Switch Announcement Message, and Operator Service. Destinations may be given as a country, carrier, or both. Each Switch Interval thus creates multiple groups of CDRs in accordance with a destination metrics and services metrics. These groups are instantiated objects comprising CDR metrics and methods for processing those CDR metrics. The objects may be referred to as carrier_service_metric objects and country_service_metric objects. For each service, a Switch Interval can place a CDR in a carrier_service_metric object, a country_service_metric object, or both, depending on user presentation and reporting requirements. For example, as shown in FIG. 4, there is a carrier_service_metric object called BT_OpSvs; representing CDRs that are created from operator service calls to British Telecom. There may be a country_service_metric object called Ger_VPN for VPN calls to Germany.

Preferably, each carrier_service_metric object contains a CDR metric object for that carrier/service combination for calculating CDR metrics. In a preferred embodiment, seizures and failures are counted. Each CDR metric object represents a peg count of CDRs that contribute to that metric. For each CDR that a carrier_service_metric object receives for a failed call, for example, the "failures" counter is incremented by one. From seizures and failures, ASR may be calculated by the carrier service metric object according to the following simple formula: [ASR=(seizures-failures)/seizures]. The metrics may be stored in the OODB 30 when the Switch Interval has been committed to the OODB 30.

A carrier_service_metric object preferably groups its CDRs by route number. A route_service_metric object may be instantiated for each route number, and each CDR may be written to the appropriate route_service_metric object. Each route_service_metric tracks CDR metrics for an individual route and service. By definition, a route only serves one destination, so no carrier designation is needed. However, a route may carry calls for multiple services, so a route_service_metric object exists for each service type carried over a particular route. Like the carrier_service_metric objects, route_service_metric objects preferably include CDR metric objects that calculate the seizures, failures, and ASR for corresponding route/service combinations, and records these metrics to the OODB 30. These calculations are preferably performed within time intervals defined by the Switch Interval groupings.

Router

Each route_service_metric object maintains a list of failed calls and failure reasons. This may be done by instantiating a CDR Detail object, or simply by keeping all CDR detail data as attributes of the route_service_metric object. Either way, the route_service_metric object writes all CDR detail data for its route/service to the OODB 30. After the relevant metrics are recorded in OODB 30, they may be graphically displayed for user evaluation. FIG. 5 illustrates exemplary internal process architecture for generating graphic displays of call metrics or GUIs.

metric database

Another feature of the invention, its presentation. The process for generating graphic displays realized by the

internal process architecture of FIG. 5 will be referred to hereinafter as Build Monitors. The graphic displays are created from the data in the OODB **30**. The graphic displays depict call metrics preferably in a grid format. In addition, in accordance with a advantageous aspect of the invention, each call metric may be color coded to indicate the level of criticality of that metric. In operation, Build Monitors extracts data from the OODB **30**, applies threshold rules to each metric to determine the level of criticality of that metric, translates the level of criticality to a color code, and sends all data and color codes to each client **12**'s GUI. The client **12**'s GUI then formats the data into screens for user presentation.

Build Monitors preferably performs this process in the same time interval by which CDR Process groups CDRs. The completion of each Switch Interval triggers a process execution by Build Monitors. Thus, all call data is collected and presented each time a single gateway Switch Interval is completed.

As shown in FIG. **5**, CDR Process commits a completed Switch Interval object to the OODB **30** upon receipt of a CDR that is timestamped X minutes beyond the Switch Interval start time. In a preferred embodiment, X=15. When CDR Process commits a completed Switch Interval object to the OODB **30**, it triggers the Build Monitors process execution.

Build Monitors extracts from the OODB **30** all call metrics. These include seizures, failures, answers, and ASRs, grouped by destination/service, route/service, failure category, and other parameters as previously described. The data are collected from each carrier_service_metric object, country_service_metric object, and route_service_metric object that is generated from each Switch Interval object. Because calls for a single destination/service group may be carried by more than one gateway switch, the Build Monitors process preferably collect calls statistics for a certain destination/service group from multiple carrier_service_metric/country_service_metric objects (one from each Switch Interval object for a gateway switch that carried calls for that destination/service group).

Build Monitors retrieves threshold objects from a Thresholds Database for carrier and country services. The Thresholds Database is preferably an OODB that embodies thresholds as objects. In a preferred embodiment, there are two thresholds for each metric for each hour of each day of the week. Further, thresholds for seizures and ASRs are preferably maintained for each hour for each day. A first threshold represents "warning" levels, which may be coded yellow, and a second threshold represents "critical" levels, which may be coded red. The thresholds are dynamic, in that they may change with each hour. Also, the thresholds may applied to mean (average) values, which are dynamically updated by the process described in connection with FIG. **6**.

FIG. **7** illustrates an internal process architecture for implementing and applying thresholds. For each carrier_service_metric or country_service_metric object, the Build Monitors process retrieves a corresponding threshold control object from the Thresholds Database. Each threshold control object includes a list of threshold objects for that destination and service combination. Each threshold object may be specific to a day/hour segment, and preferably includes the methods for applying the thresholds. An historical mean for each metric may also be included in the threshold object. Actual threshold values, which in the preferred embodiment are given as percentages of variance from a mean, may be maintained in a configurable file and may be loaded from that file into the threshold object in the threshold database **40**.

In operation, the Build Monitors process calls on a threshold control object and passes to it the CDR metric values from the carrier_service_metric or country_service_metric object. The threshold control object then calls on the threshold object that corresponds to the particular day/hour in which the switch interval falls. The threshold object then retrieves threshold values from the database and applies them to the CDR metrics. For example, the threshold object applies two thresholds (one for warning level and one for critical level) to each of two metrics (seizures and ASR) to determine with what color code each of these metrics should be presented. The threshold object then returns to Build Monitors the value and color code for each metric.

Once the thresholds are calculated they are preferably applied. A preferred method of applying thresholds is described as follows. This thresholds process compares the seizures and ASR values of the actual data that is collected from the OODB **30** with the thresholds to determine the color of each cell in a graphical display. An historical mean value is used for each metric, which is drawn from historical data in the OODB **30**. Mean values are dynamic, they are constantly updated to reflect changing data. In a preferred embodiment, the mean is calculated over the most recent 8 samples. Thresholds may be stated as a percent delta from the mean. Though thresholds are given on a per hour basis, they are applied to regular intervals, e.g., 15-minute intervals, of CDR metrics. Thus, an hourly threshold may be compared to four intervals of data for a metric.

EXAMPLE 1

Seizures

Suppose a mean value is given as 800 (indicating 800 seizures for a 15 minute interval). A warning threshold for a day/hour interval may be given as 20% delta; this means the threshold is exceeded if the actual value of seizures for a 15-minute interval is 20% greater or less than the mean value.

A critical threshold for a day/hour interval is given as 30% delta; this means the threshold is exceeded if the actual value of seizures for a 15-minute interval is 30% greater or less than the mean value. Thus, the seizures value is color-coded: white, if X<20% delta from 800; yellow, if 20%<X<30% delta from 800; red, if X>30% delta from 800; 20% of 800=160; 30% of 800=240. If the actual number of seizures, X, for the 15-minute interval that falls within this day/hour interval is between: 959 and 641, the "seizures" value is coded white; 561 and 640, or 960 and 1039, the value is coded yellow; 0 and 560, or greater than 1040, the value is coded red.

EXAMPLE 2

ASR

Suppose a mean value is given as 70 (ASR is expressed as a percent). A warning threshold for a day/hour interval may be given as 20% below mean.

A critical threshold for a day/hour interval may be given as 30% or more below mean. (With ASR, we are only concerned if it falls too low.) Thus, the ASR value is color-coded: white, if ASR is greater than, 70–20%=56; yellow, if ASR is between: 70–29%=50 and 56, inclusive; red, if ASR is equal to or less than: 70–30%=49.

All of the values given in Examples 1 and 2 are for purposes of illustration only. The skilled artisan will recog-

nize that the values may be changed to accommodate changing conditions.

Build Monitors stores data (actual values and color codes) in a Presentation Database **42**. Data may be stored as flat files in a relational database or as objects in an OODB, or as other types of data in other types of databases. The Presentation Database **42** may be located on the server **10**, on a different but co-located computer, or, preferably on a remote computer that is local to the clients **12** for performance purposes.

When the Presentation Database **42** has been built for the latest interval, Build Monitors sends a message to each Client **12** that is logged in to the server **10**. This message triggers the clients **12** to retrieve the latest data from the Presentation Database **42**. Each Client **12** retrieves and displays the latest data from the Presentation Database **42**. In this way, the graphic displays are updated automatically, without the user having to trigger a screen update. Each object in the Presentation Database **42** (or file if it is a relational database) has all data calculated by the carrier_service_metric and route_service_metric processes. When a user selects an ASR value from the Inbound Services screen (not shown) for example, the Client **12** retrieves from the Presentation Database **42** the route-by-route ASR values calculated by the route_service_metric objects. When a user selects a particular route, the Client **12** retrieves from the Presentation Database **42** the ASR values for each dialed number prefix and/or failure category.

FIG. **6** illustrates a process architecture for dynamically updating thresholds in accordance with the invention. This process is referred to henceforth as Update Thresholds. This process may be executed by the server **10** independently of the main process thread. Thus, it does not intrude on the CDR Process or Build Monitors, which are time-sensitive. With reference to FIG. **6**, Update Thresholds monitors CDR Process to detect when the main process thread is idle enough so as not to engender interference. When the Build Monitors is finished with the current switch intervals, the switch intervals will be marked obsolete so as to avoid contention with current metrics.

At such a time, Update Thresholds collects from the OODB **30** currently recorded metrics for each interval for each switch. Update Thresholds then calculates the mean value for each metric. In a preferred embodiment, the mean value of each metric is calculated utilizing the most recent 8 samples for each hour of each day of the week. This calculation may be performed using an algorithm that weights the previous historical mean value with a factor of 7/8 and weights the current value with a factor of 1/8. Update Thresholds then updates the Thresholds Database with these new mean values.

INDUSTRIAL APPLICABILITY

This invention may be used in connection with any network that generates CDRs. Accordingly, the invention is not limited by the physical topology or construction of any network. More particularly, the invention may be used with conventional circuit-switched networks as well as packet-switched networks, such as the Internet.

The invention is particularly suited to international circuit-switched networks. FIG. **8** illustrates an exemplary international circuit-switched network. Switches A, B and C interconnect the domestic switching network **50** with foreign switches **52–58+**. Thus, switches A, B and C may be referred to as gateway switches. Examples of foreign carrier switches are Switch ME (Mercury) and Switch BT (British Telecom), both operating in the UK; Switch DT (Deutsche Telecom) operating in Germany; and Switch AV (Avantel), operating in Mexico. Particular to note is that there may be multiple routes between the same two switches, multiple routes being distinguished by different physical transmission routes. Also, multiple gateway switches (i.e., Switch A and Switch B) may be used to interconnect with the same foreign carrier switch. Also, there are transit routes, in which an intermediate carrier (Switch BT) handles calls that are destined for another country (Switch DT). With respect to the present invention, service impacts are reported based on both the final destination of calls (i.e., Switch DT) and the transit destination (Switch BT).

While various embodiments of the present invention have been described, it should be understood that they have been presented by way of example only, and not limitation. The present invention is not limited to any particular service, or destination type or route type. Services and routes are simply codes obtained from CDRs and used to group CDRs. Destinations are derived from CDR data, and are also used to group CDRs. Additional modifications and variations of the described embodiments within the scope of the appended claims will be apparent to the skilled artisan.

What is claimed is:

1. A method for analyzing the effects of network events on call services comprising:

receiving a plurality of call detail records (CDRs) from a plurality of network devices, each of the plurality of CDRs including a time stamp, call service data and call destination data;

grouping the plurality of CDRs into a plurality of first subgroups according to corresponding network devices and to the time stamps of the plurality of CDRs;

for each of the plurality of first subgroups, grouping the CDRs into a plurality of second subgroups according to a call service data and call destination data; and

generating call statistics indicative of a state of call services from data contained in the CDRs for each of the plurality of second subgroups.

2. An apparatus for analyzing the effect of network events on call services comprising:

means for receiving a plurality of call detail records (CDRs) from a plurality of network devices, each of the plurality of CDRs including a time stamp, a call service data and a call destination data;

means for grouping the plurality of CDRs into a plurality of first subgroups according to corresponding network devices and to the time stamps of the plurality of CDRs;

means for grouping the CDR's of each of the plurality of first subgroups into a plurality of second subgroups according to a call service data and call destination data; and

means for generating call statistics indicative of a state of call services from data contained in the CDRs for each of the plurality of second subgroups.

3. An apparatus for analyzing the effect of network events on call services, said apparatus including a computer programmed to:

receive a plurality of call detail records (CDRs) from a plurality of network devices, each of the plurality of CDRs including a time stamp, call service data and call destination data;

group the plurality of CDRs into a plurality of first subgroups according to corresponding network devices and to the time stamps of the plurality of CDRs;

for each of the plurality of first subgroups, to group the CDRs into a plurality of second subgroups according to a call service data and call destination data; and

generate call statistics indicative of a state of call services from data contained in the CDRs for each of the plurality of second subgroups.

4. A system for analyzing the effect of network events on call services comprising:

means for receiving a plurality of call detail records (CDRs) from a plurality of network devices, each of the plurality of CDRs including a time stamp, a call service data and a call destination data;

means for grouping the plurality of CDRs into a plurality of first subgroups according to corresponding network devices and to the time stamps of the plurality of CDRs;

means for grouping the CDR's of each of the plurality of first subgroups into a plurality of second subgroups according to a call service data and call destination data;

means for generating call statistics from metrics contained in the CDRs for each of the plurality of second subgroups; and

means for displaying the statistics in a user friendly easily readable manner.

5. A method for analyzing the effects of network events on call services comprising:

receiving a plurality of call detail records (CDRs) from a plurality of network devices, each of the plurality of CDRs including a time stamp, call service data and call destination data;

grouping the plurality of CDRs into a plurality of first subgroups according to corresponding network devices and to the time stamps of the plurality of CDRs;

for each of the plurality of first subgroups, grouping the CDRs into a plurality of second subgroups according to a call service data and call destination data;

generating call statistics indicative of a state of call services from data contained in the CDRs for each of the plurality of second subgroups;

generating a threshold value for each call statistic, each threshold value including a dynamic mean value of a corresponding call statistic over a selected time period;

comparing each call statistic to a corresponding threshold value and visually coding the statistics responsive to the comparison; and

displaying the visually coded statistics to indicate call service alarm conditions.

* * * * *

US006535227B1

(12) United States Patent
Fox et al.

(10) Patent No.: US 6,535,227 B1

(45) Date of Patent: Mar. 18, 2003

(54) SYSTEM AND METHOD FOR ASSESSING THE SECURITY POSTURE OF A NETWORK AND HAVING A GRAPHICAL USER INTERFACE

(75) Inventors: Kevin L. Fox, Palm Bay, FL (US); Ronda R. Henning, West Melbourne, FL (US); John T. Farrell, Melbourne, FL (US); Clifford C. Miller, Palm Bay, FL (US)

(73) Assignee: Harris Corporation, Melbourne, FL (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: 09/500,101

(22) Filed: Feb. 8, 2000

(51) Int. Cl.[7] G09G 5/00; G06F 15/173; G06F 15/16

(52) U.S. Cl. 345/736; 709/224; 709/223; 713/201

(58) Field of Search 345/734, 735, 345/736, 743, 741, 742, 764, 772, 969; 709/223, 224; 713/201

(56) References Cited

U.S. PATENT DOCUMENTS

5,138,321 A 8/1992 Hammer 342/36
5,485,409 A 1/1996 Gupta et al. 395/186
5,684,957 A 11/1997 Kondo et al. 395/200.06
5,699,403 A 12/1997 Ronnen 379/32
5,745,382 A 4/1998 Vilim et al. 364/551.01
5,751,965 A * 5/1998 Mayo et al. 345/734
5,764,913 A * 6/1998 Jancke et al. 345/764
5,768,552 A * 6/1998 Jacoby 345/418
5,787,235 A 7/1998 Smith et al. 395/50
5,798,939 A 8/1998 Ochoa et al. 364/493
5,812,763 A 9/1998 Teng 395/187.01
5,892,903 A 4/1999 Klaus 395/187.01
5,963,653 A 10/1999 McNary et al. 382/103
6,020,889 A * 2/2000 Tarbox et al. 345/736
6,271,845 B1 * 8/2001 Richardson 345/969
6,330,005 B1 * 12/2001 Tonelli et al. 345/734
6,356,282 B2 * 3/2002 Roytman et al. 345/736

FOREIGN PATENT DOCUMENTS

EP 0 961 440 A2 12/1999 .... H04L/12/24
WO WO 99/56195 11/1999 .... G06F/1/00

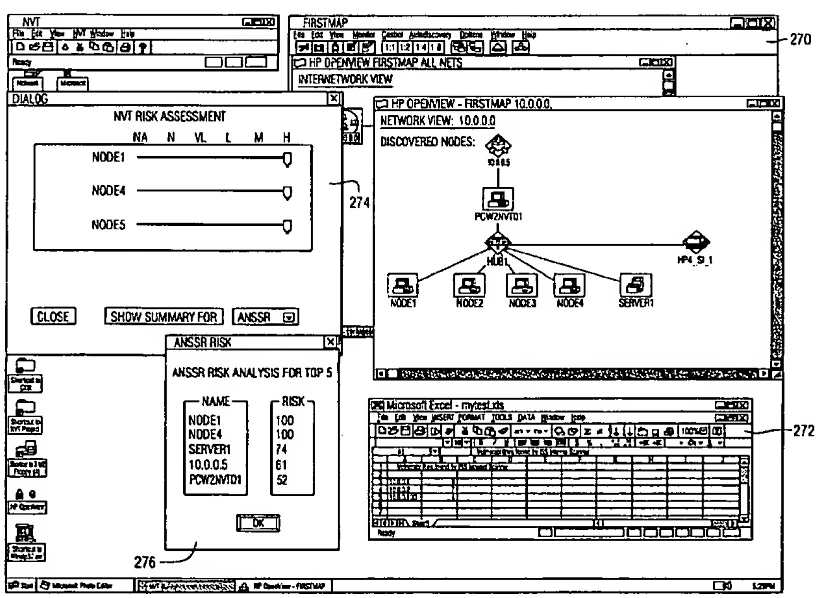
* cited by examiner

Primary Examiner—Kristine Kincaid
Assistant Examiner—Thanh Vu
(74) Attorney, Agent, or Firm—Allen, Dyer, Doppelt, Milbrath & Gilchrist, P.A.

(57) ABSTRACT

A graphical user interface is contained on a computer screen and used for determining the vulnerability posture of a network. A system design window displays network items of a network map that are representative of different network elements contained within the network. The respective network icons are linked together in an arrangement corresponding to how network elements are interconnected within the network. Selected portions of the network map turn a different color indicative of a vulnerability that has been established for that portion of the network after a vulnerability posture of the network has been established.

24 Claims, 14 Drawing Sheets



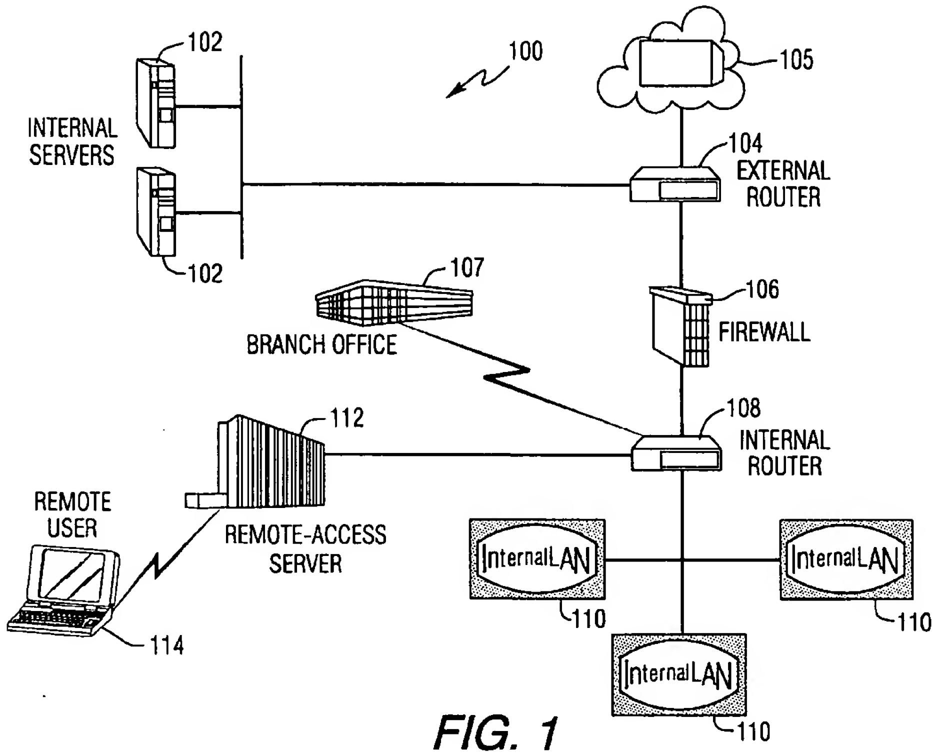

100
102
INTERNAL
SERVERS
102
105
104
EXTERNAL
ROUTER
107
BRANCH OFFICE
106
FIREWALL
112
108
INTERNAL
ROUTER
REMOTE
USER
REMOTE-ACCESS
SERVER
InternalLAN
InternalLAN
InternalLAN
110
110
110
114

FIG. 1

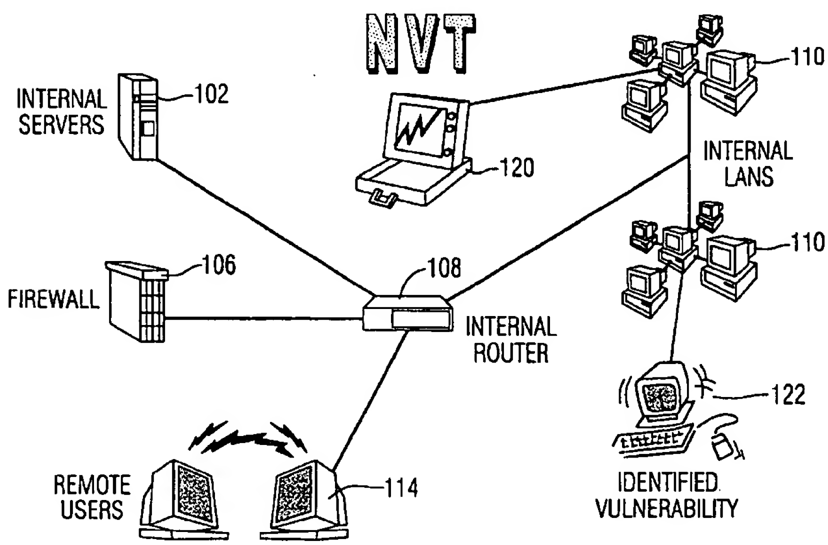

NVT
110
INTERNAL
SERVERS
102
120
INTERNAL
LANS
110
106
FIREWALL
108
INTERNAL
ROUTER
122
REMOTE
USERS
114
IDENTIFIED
VULNERABILITY

FIG. 2

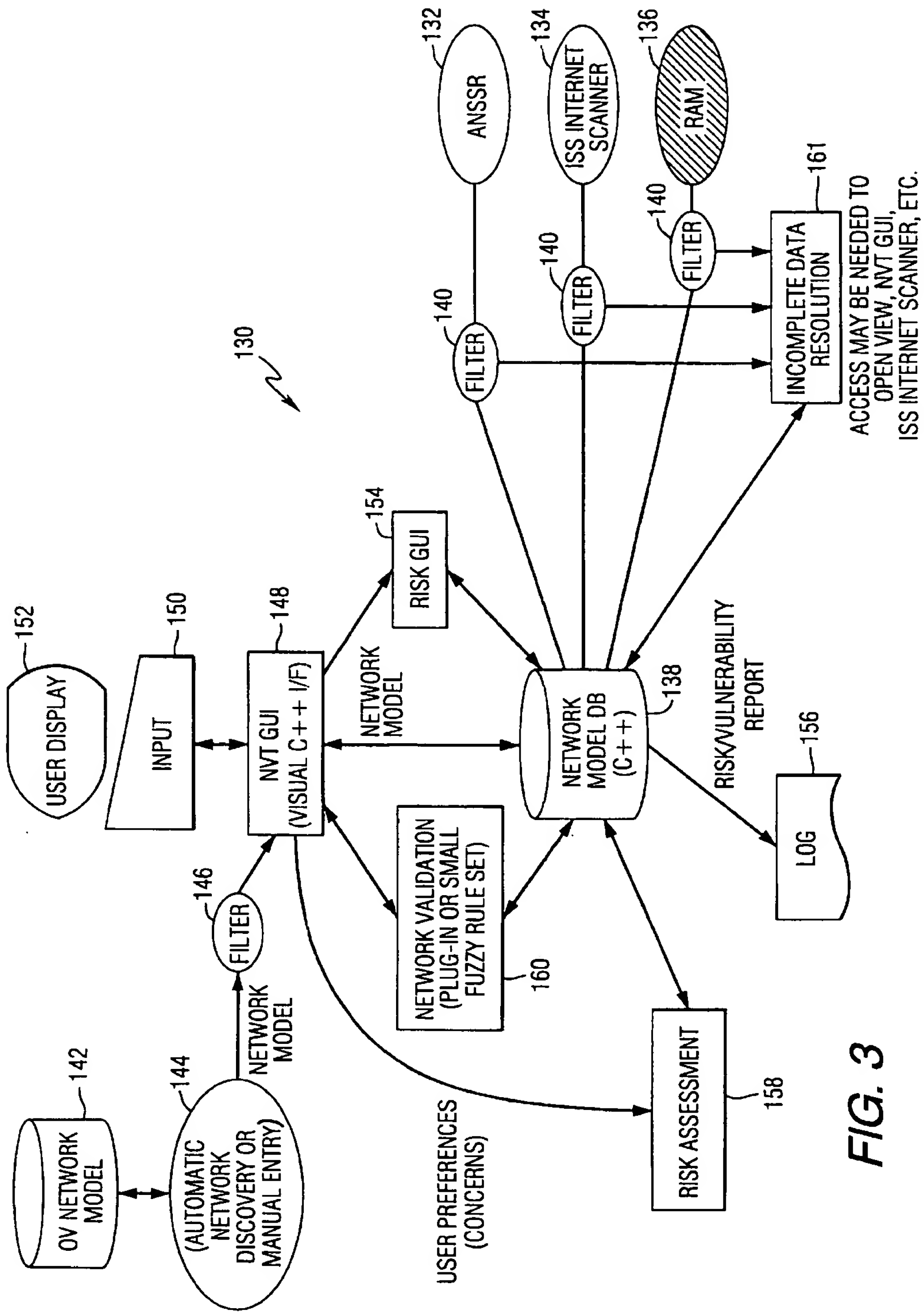

130
ANSSR
132
ISS INTERNET SCANNER
134
RAM
136
FILTER
140
FILTER
140
FILTER
140
INCOMPLETE DATA RESOLUTION
161
ACCESS MAY BE NEEDED TO OPEN VIEW, NVT GUI, ISS INTERNET SCANNER, ETC.
USER DISPLAY
152
INPUT
150
NVT GUI (VISUAL C++ I/F)
148
RISK GUI
154
NETWORK MODEL
NETWORK MODEL DB (C++)
138
RISK/VULNERABILITY REPORT
LOG
156
FILTER
146
NETWORK VALIDATION (PLUG-IN OR SMALL FUZZY RULE SET)
160
OV NETWORK MODEL
142
(AUTOMATIC NETWORK DISCOVERY OR MANUAL ENTRY)
144
NETWORK MODEL
USER PREFERENCES (CONCERNS)
RISK ASSESSMENT
158

FIG. 3

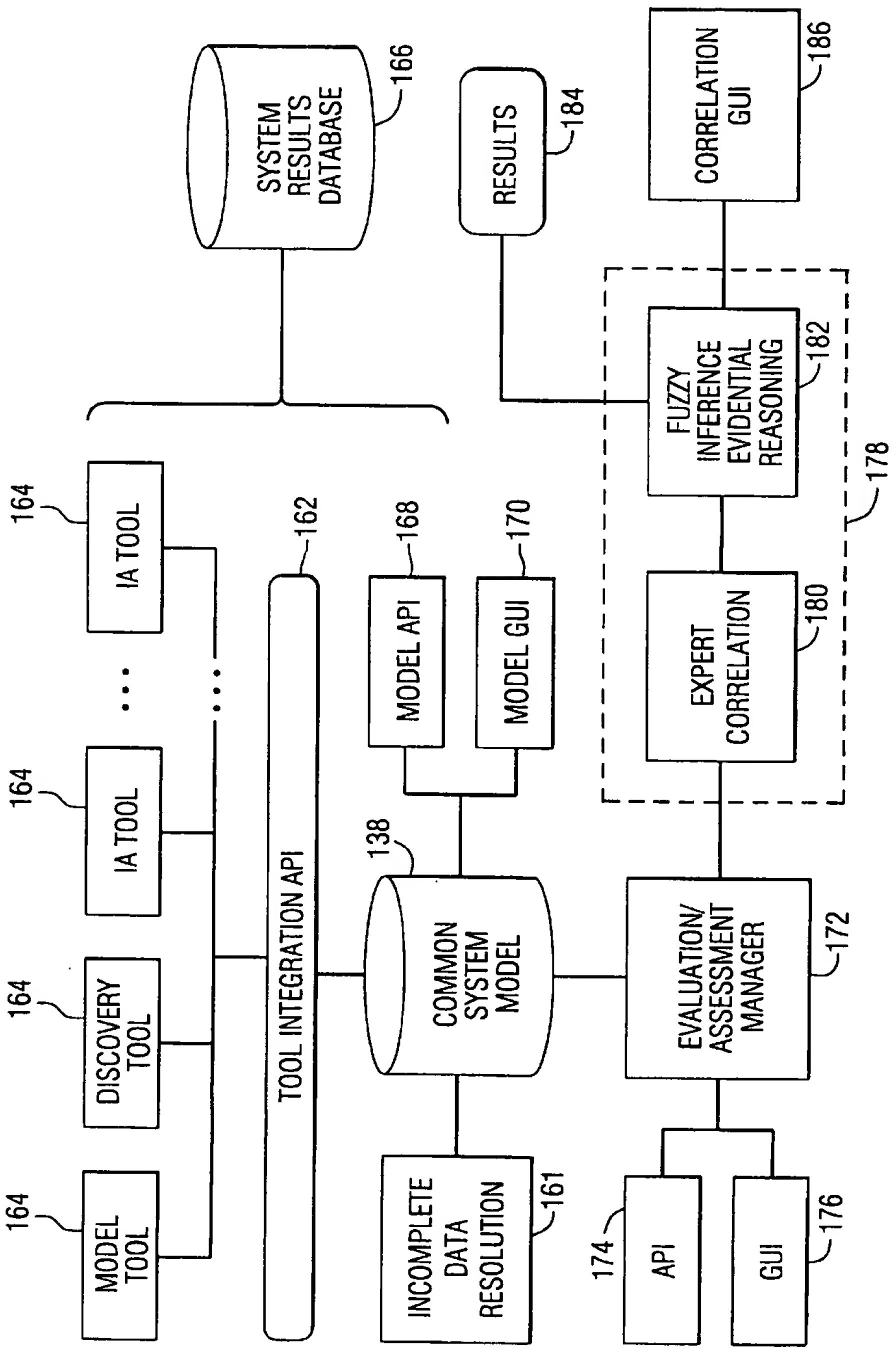

SYSTEM RESULTS DATABASE
166
RESULTS
184
CORRELATION GUI
186
IA TOOL
164
IA TOOL
164
DISCOVERY TOOL
164
MODEL TOOL
164
TOOL INTEGRATION API
162
MODEL API
168
MODEL GUI
170
COMMON SYSTEM MODEL
138
INCOMPLETE DATA RESOLUTION
161
FUZZY INFERENCE EVIDENTIAL REASONING
182
EXPERT CORRELATION
180
178
EVALUATION/ ASSESSMENT MANAGER
172
API
174
GUI
176

FIG. 4

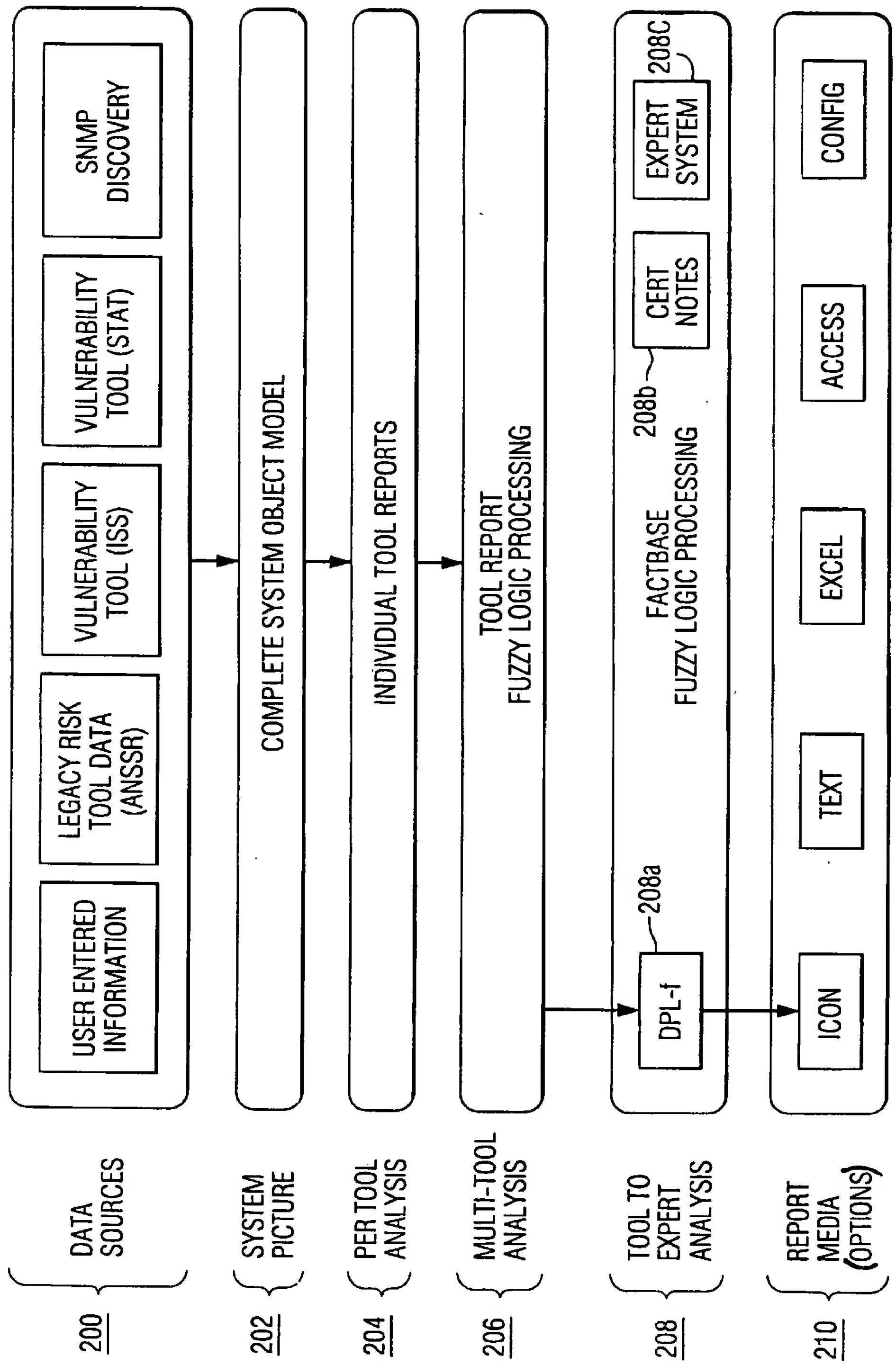
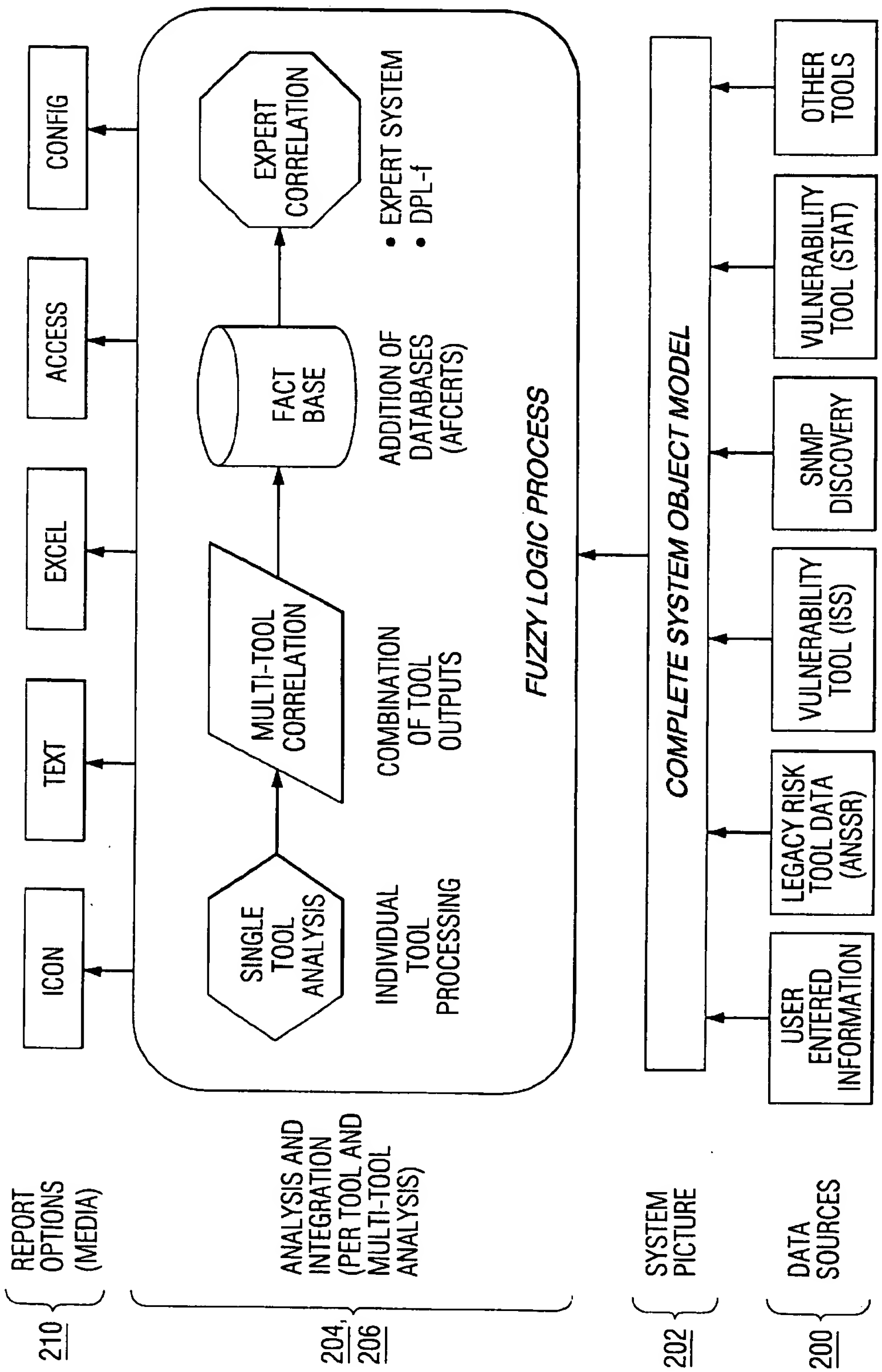

USER ENTERED INFORMATION
LEGACY RISK TOOL DATA (ANSSR)
VULNERABILITY TOOL (ISS)
VULNERABILITY TOOL (STAT)
SNMP DISCOVERY
COMPLETE SYSTEM OBJECT MODEL
INDIVIDUAL TOOL REPORTS
TOOL REPORT FUZZY LOGIC PROCESSING
DPL-f
208a
FACTBASE FUZZY LOGIC PROCESSING
208b
CERT NOTES
EXPERT SYSTEM
208C
ICON
TEXT
EXCEL
ACCESS
CONFIG
DATA SOURCES
SYSTEM PICTURE
PER TOOL ANALYSIS
MULTI-TOOL ANALYSIS
TOOL TO EXPERT ANALYSIS
REPORT MEDIA (OPTIONS)
200
202
204
206
208
210

FIG. 5



210
REPORT OPTIONS (MEDIA)
ICON
TEXT
EXCEL
ACCESS
CONFIG
204, 206
ANALYSIS AND INTEGRATION (PER TOOL AND MULTI-TOOL ANALYSIS)
SINGLE TOOL ANALYSIS
INDIVIDUAL TOOL PROCESSING
MULTI-TOOL CORRELATION
COMBINATION OF TOOL OUTPUTS
FACT BASE
ADDITION OF DATABASES (AFCERTS)
EXPERT CORRELATION
• EXPERT SYSTEM
• DPL-f
FUZZY LOGIC PROCESS
202
SYSTEM PICTURE
COMPLETE SYSTEM OBJECT MODEL
200
DATA SOURCES
USER ENTERED INFORMATION
LEGACY RISK TOOL DATA (ANSSR)
VULNERABILITY TOOL (ISS)
SNMP DISCOVERY
VULNERABILITY TOOL (STAT)
OTHER TOOLS

FIG. 6

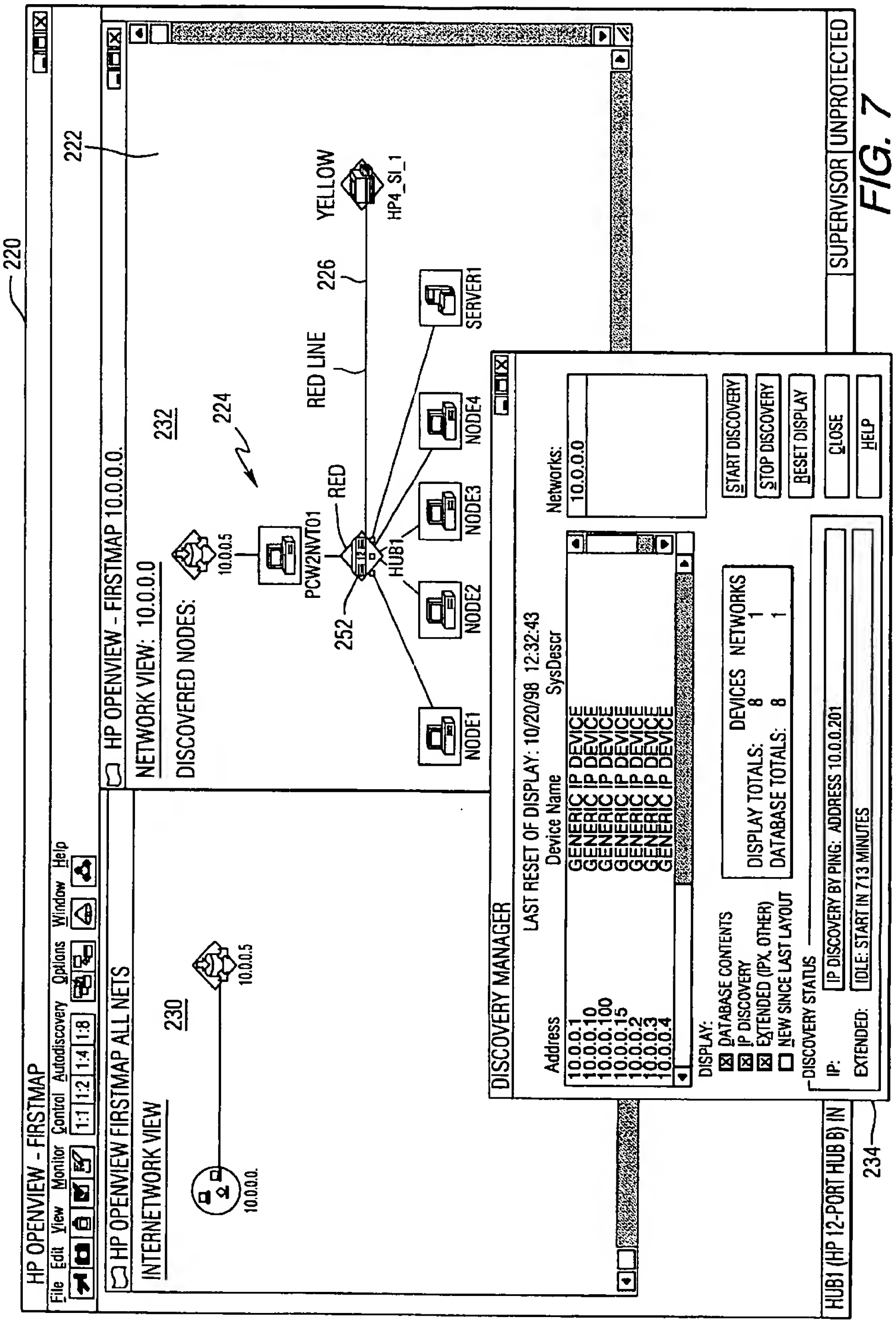

HP OPENVIEW - FIRSTMAP
File Edit View Monitor Control Autodiscovery Options Window Help
1:1 1:2 1:4 1:8
HP OPENVIEW FIRSTMAP ALL NETS
INTERNETWORK VIEW
230
10.0.0.0.
10.0.0.5
HP OPENVIEW - FIRSTMAP 10.0.0.0.
NETWORK VIEW: 10.0.0.0
DISCOVERED NODES:
232
224
10.0.0.5
PCW2NVT01
RED
252
HUB1
RED LINE
226
YELLOW
HP4_SI_1
NODE1
NODE2
NODE3
NODE4
SERVER1
220
222
DISCOVERY MANAGER
LAST RESET OF DISPLAY: 10/20/98 12:32:43
Address Device Name SysDescr
10.0.0.1 GENERIC IP DEVICE
10.0.0.10 GENERIC IP DEVICE
10.0.0.100 GENERIC IP DEVICE
10.0.0.15 GENERIC IP DEVICE
10.0.0.2 GENERIC IP DEVICE
10.0.0.3 GENERIC IP DEVICE
10.0.0.4 GENERIC IP DEVICE
Networks:
10.0.0.0
DISPLAY:
DATABASE CONTENTS
IP DISCOVERY
EXTENDED (IPX, OTHER)
NEW SINCE LAST LAYOUT
DEVICES NETWORKS
DISPLAY TOTALS: 8 1
DATABASE TOTALS: 8 1
START DISCOVERY
STOP DISCOVERY
RESET DISPLAY
CLOSE
HELP
DISCOVERY STATUS
IP: IP DISCOVERY BY PING: ADDRESS 10.0.0.201
EXTENDED: IDLE: START IN 713 MINUTES
HUB1 (HP 12-PORT HUB B) IN
234
SUPERVISOR UNPROTECTED

FIG. 7

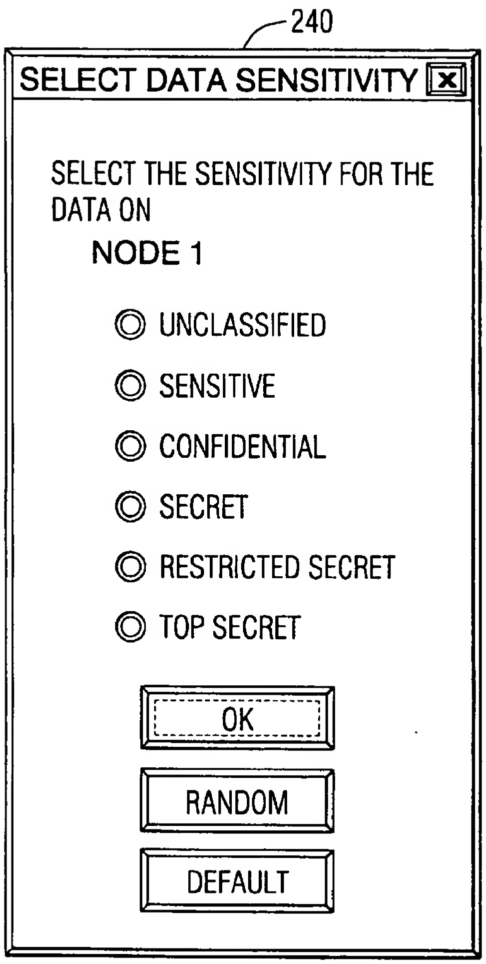

240
SELECT DATA SENSITIVITY
SELECT THE SENSITIVITY FOR THE DATA ON
NODE 1
UNCLASSIFIED
SENSITIVE
CONFIDENTIAL
SECRET
RESTRICTED SECRET
TOP SECRET
OK
RANDOM
DEFAULT

FIG. 8A

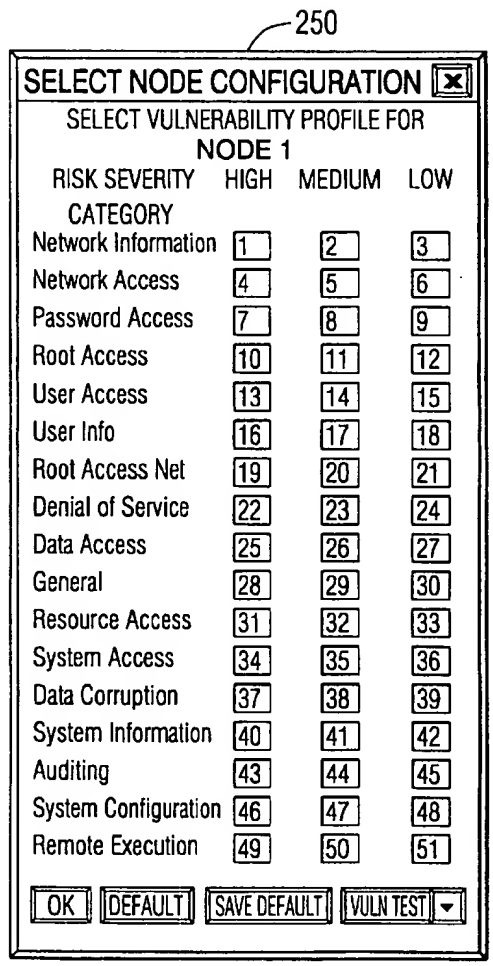

250
SELECT NODE CONFIGURATION
SELECT VULNERABILITY PROFILE FOR
NODE 1
RISK SEVERITY CATEGORY
HIGH
MEDIUM
LOW
Network Information 1 2 3
Network Access 4 5 6
Password Access 7 8 9
Root Access 10 11 12
User Access 13 14 15
User Info 16 17 18
Root Access Net 19 20 21
Denial of Service 22 23 24
Data Access 25 26 27
General 28 29 30
Resource Access 31 32 33
System Access 34 35 36
Data Corruption 37 38 39
System Information 40 41 42
Auditing 43 44 45
System Configuration 46 47 48
Remote Execution 49 50 51
OK
DEFAULT
SAVE DEFAULT
VULN TEST

FIG. 8B

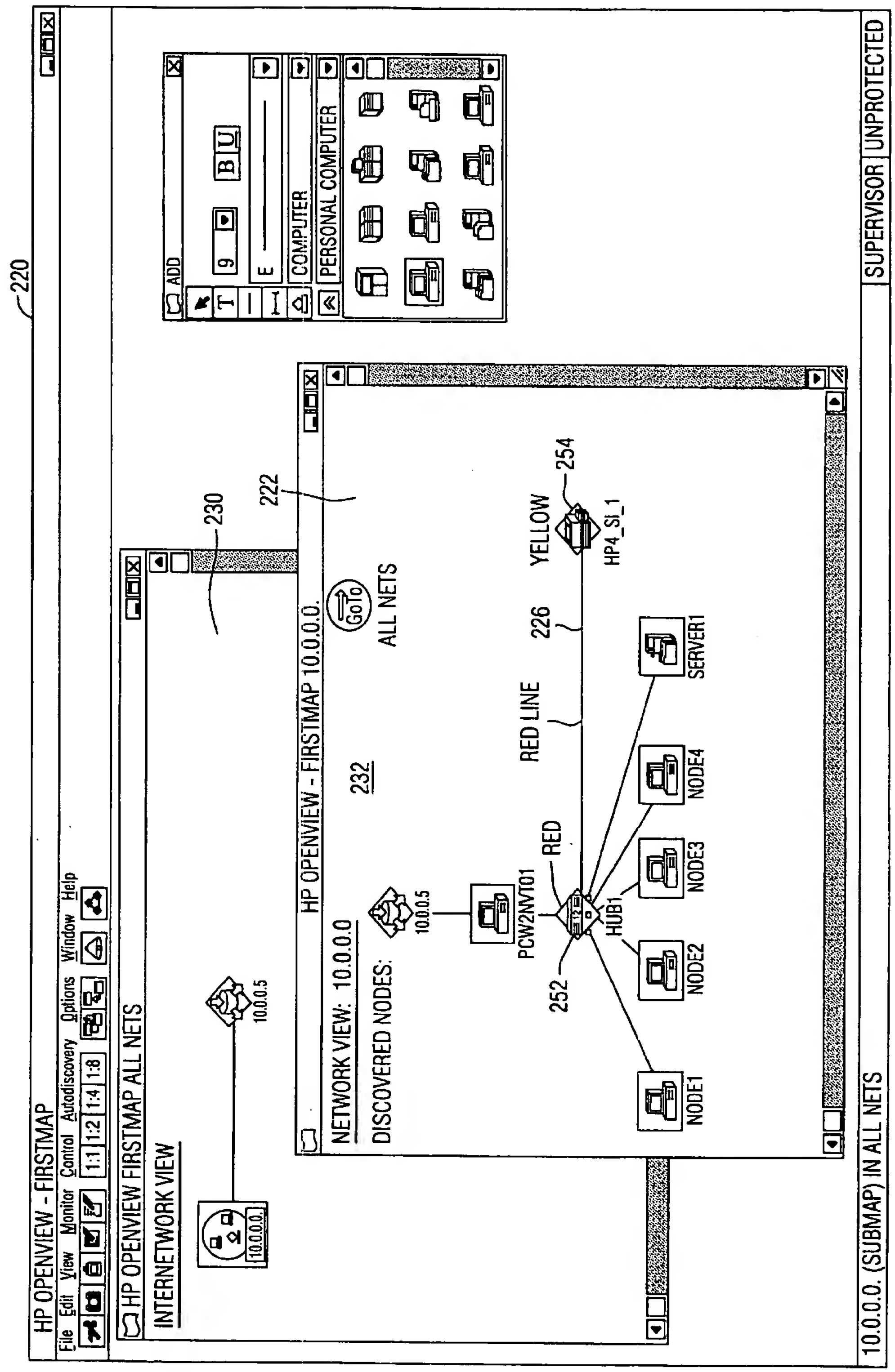

220
HP OPENVIEW - FIRSTMAP
File Edit View Monitor Control Autodiscovery Options Window Help
1:1 1:2 1:4 1:8
HP OPENVIEW FIRSTMAP ALL NETS
INTERNETWORK VIEW
10.0.0.0.
10.0.0.5
230
222
HP OPENVIEW - FIRSTMAP 10.0.0.0.
NETWORK VIEW: 10.0.0.0
DISCOVERED NODES:
232
GoTo
ALL NETS
10.0.0.5
PCW2NVT01
RED
252
HUB1
RED LINE
226
YELLOW
254
HP4_SI_1
NODE1
NODE2
NODE3
NODE4
SERVER1
ADD
T
COMPUTER
PERSONAL COMPUTER
B U
10.0.0.0. (SUBMAP) IN ALL NETS
SUPERVISOR UNPROTECTED

FIG. 9

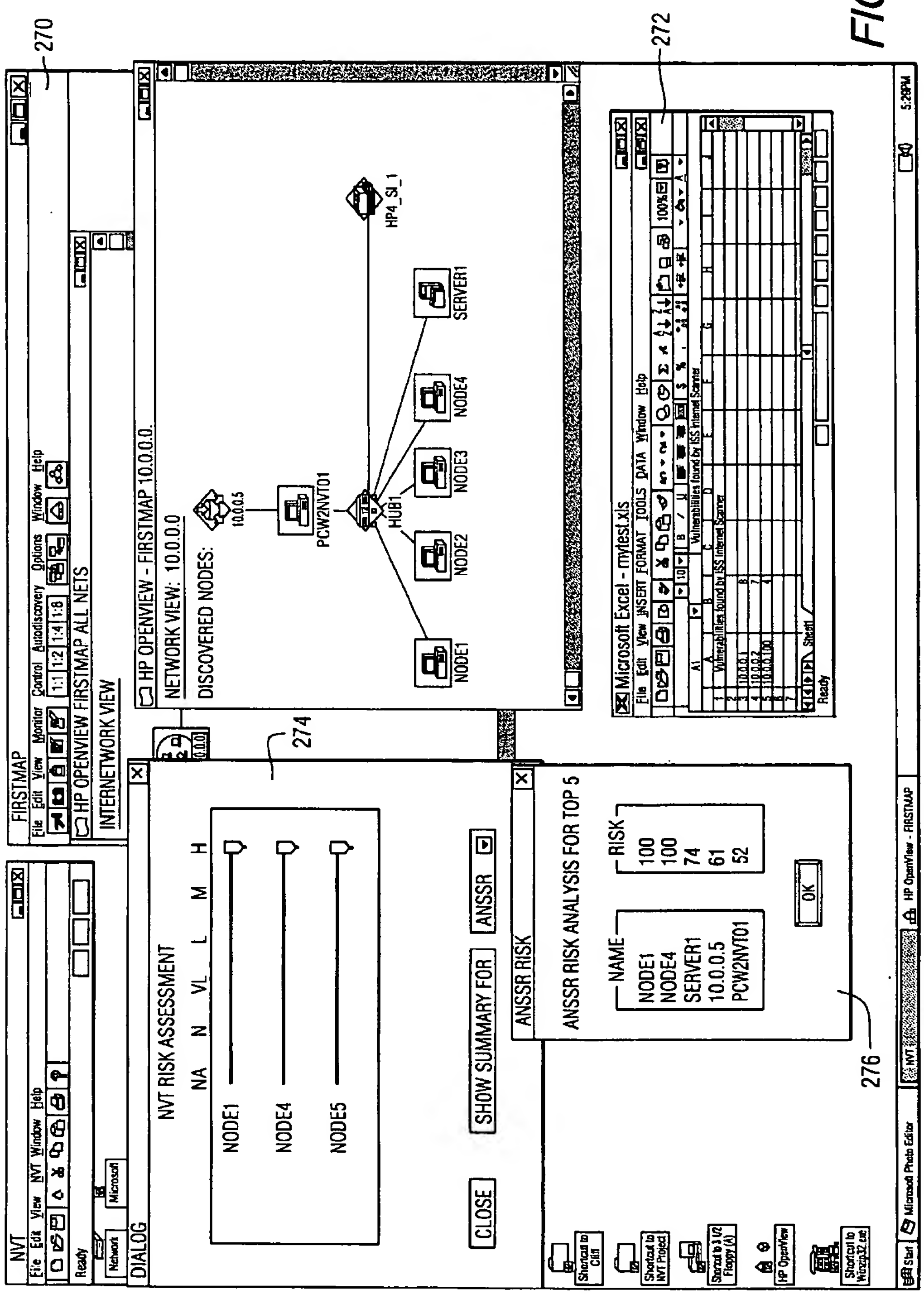

270
272
274
276
FIRSTMAP
HP OPENVIEW FIRSTMAP ALL NETS
INTERNETWORK VIEW
HP OPENVIEW - FIRSTMAP 10.0.0.0.
NETWORK VIEW: 10.0.0.0
DISCOVERED NODES:
10.0.0.5
PCW2NVT01
HUB1
HP4_SI_1
NODE1
NODE2
NODE3
NODE4
SERVER1
Microsoft Excel - mytest.xls
NVT
DIALOG
NVT RISK ASSESSMENT
NA N VL L M H
NODE1
NODE4
NODE5
SHOW SUMMARY FOR
ANSSR
CLOSE
ANSSR RISK
ANSSR RISK ANALYSIS FOR TOP 5
NAME
RISK
NODE1 100
NODE4 100
SERVER1 74
10.0.0.5 61
PCW2NVT01 52
OK

FIG. 10

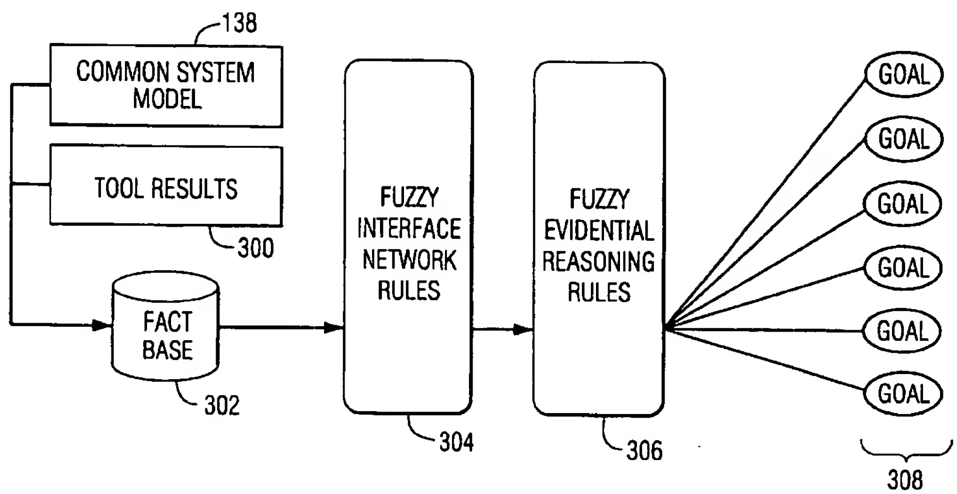

138
COMMON SYSTEM MODEL
TOOL RESULTS
300
FACT BASE
302
FUZZY INTERFACE NETWORK RULES
304
FUZZY EVIDENTIAL REASONING RULES
306
GOAL
GOAL
GOAL
GOAL
GOAL
GOAL
308

FIG. 11

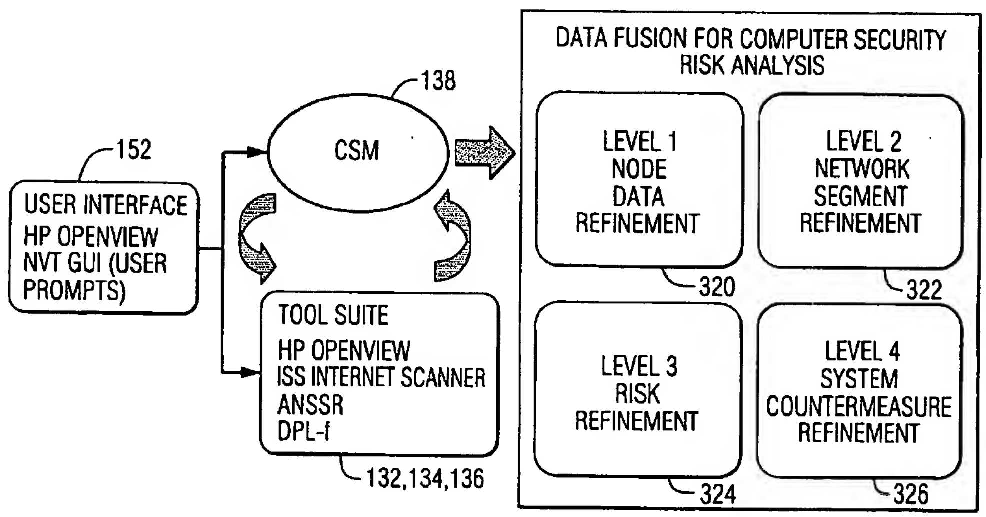

152
USER INTERFACE
HP OPENVIEW
NVT GUI (USER PROMPTS)
138
CSM
TOOL SUITE
HP OPENVIEW
ISS INTERNET SCANNER
ANSSR
DPL-f
132,134,136
DATA FUSION FOR COMPUTER SECURITY RISK ANALYSIS
LEVEL 1 NODE DATA REFINEMENT
320
LEVEL 2 NETWORK SEGMENT REFINEMENT
322
LEVEL 3 RISK REFINEMENT
324
LEVEL 4 SYSTEM COUNTERMEASURE REFINEMENT
326

FIG. 12

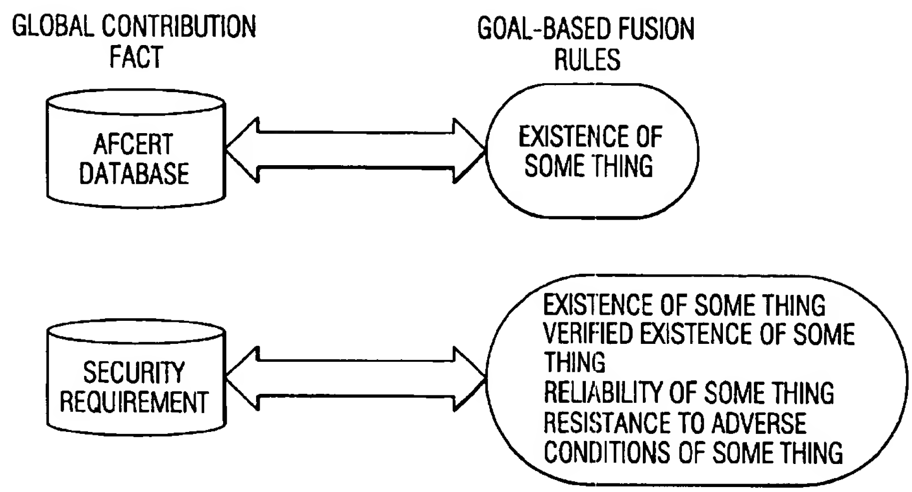

GLOBAL CONTRIBUTION FACT
GOAL-BASED FUSION RULES
AFCERT DATABASE
EXISTENCE OF SOME THING
SECURITY REQUIREMENT
EXISTENCE OF SOME THING
VERIFIED EXISTENCE OF SOME THING
RELIABILITY OF SOME THING
RESISTANCE TO ADVERSE CONDITIONS OF SOME THING

FIG. 13

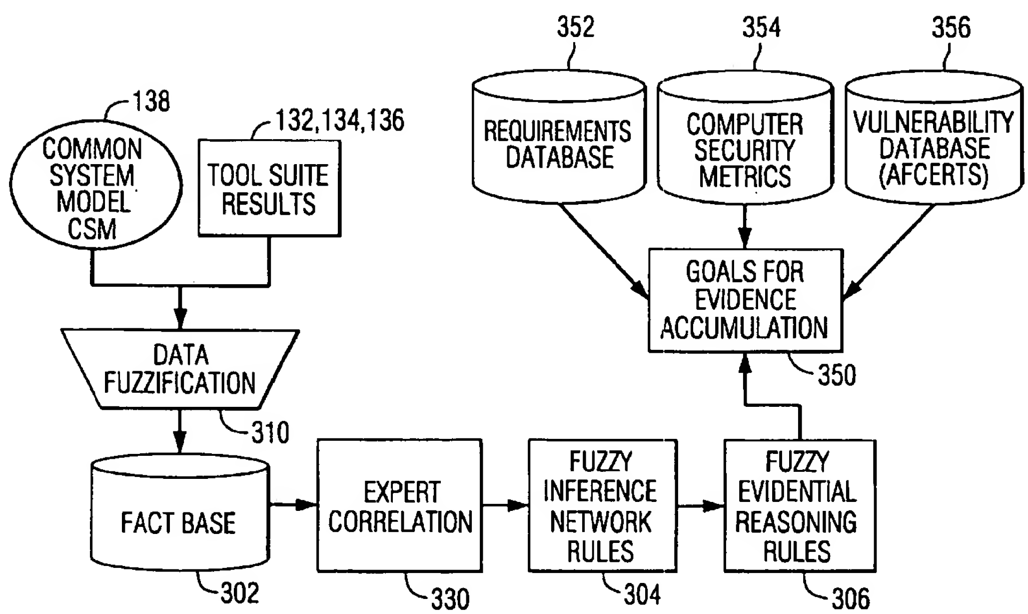

352
354
356
138
132,134,136
COMMON SYSTEM MODEL CSM
TOOL SUITE RESULTS
REQUIREMENTS DATABASE
COMPUTER SECURITY METRICS
VULNERABILITY DATABASE (AFCERTS)
GOALS FOR EVIDENCE ACCUMULATION
350
DATA FUZZIFICATION
310
FACT BASE
302
EXPERT CORRELATION
330
FUZZY INFERENCE NETWORK RULES
304
FUZZY EVIDENTIAL REASONING RULES
306

FIG. 14

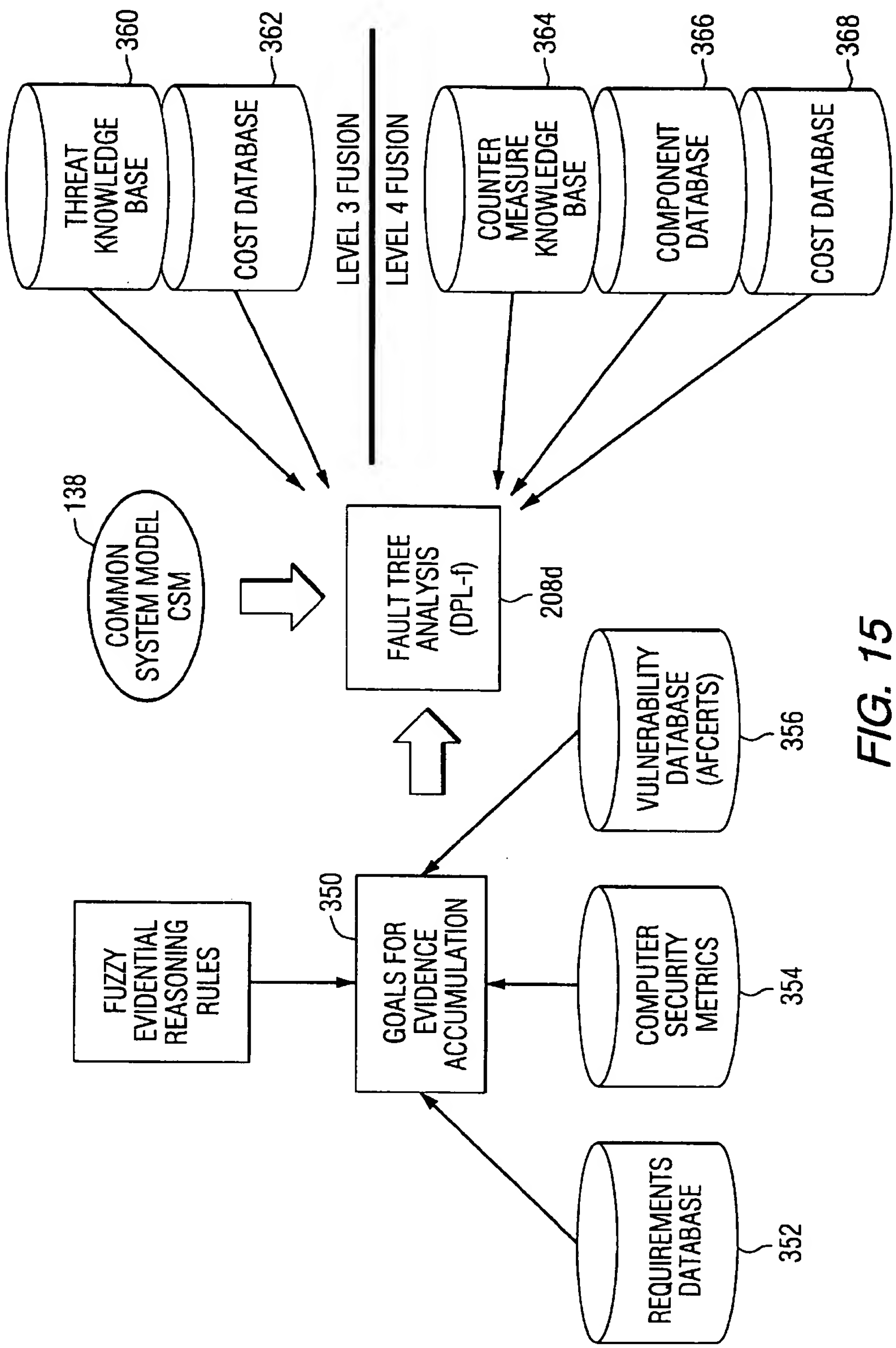

THREAT KNOWLEDGE BASE
360
COST DATABASE
362
LEVEL 3 FUSION
LEVEL 4 FUSION
COUNTER MEASURE KNOWLEDGE BASE
364
COMPONENT DATABASE
366
COST DATABASE
368
COMMON SYSTEM MODEL CSM
138
FAULT TREE ANALYSIS (DPL-f)
208d
VULNERABILITY DATABASE (AFCERTS)
356
FUZZY EVIDENTIAL REASONING RULES
GOALS FOR EVIDENCE ACCUMULATION
350
COMPUTER SECURITY METRICS
354
REQUIREMENTS DATABASE
352

FIG. 15

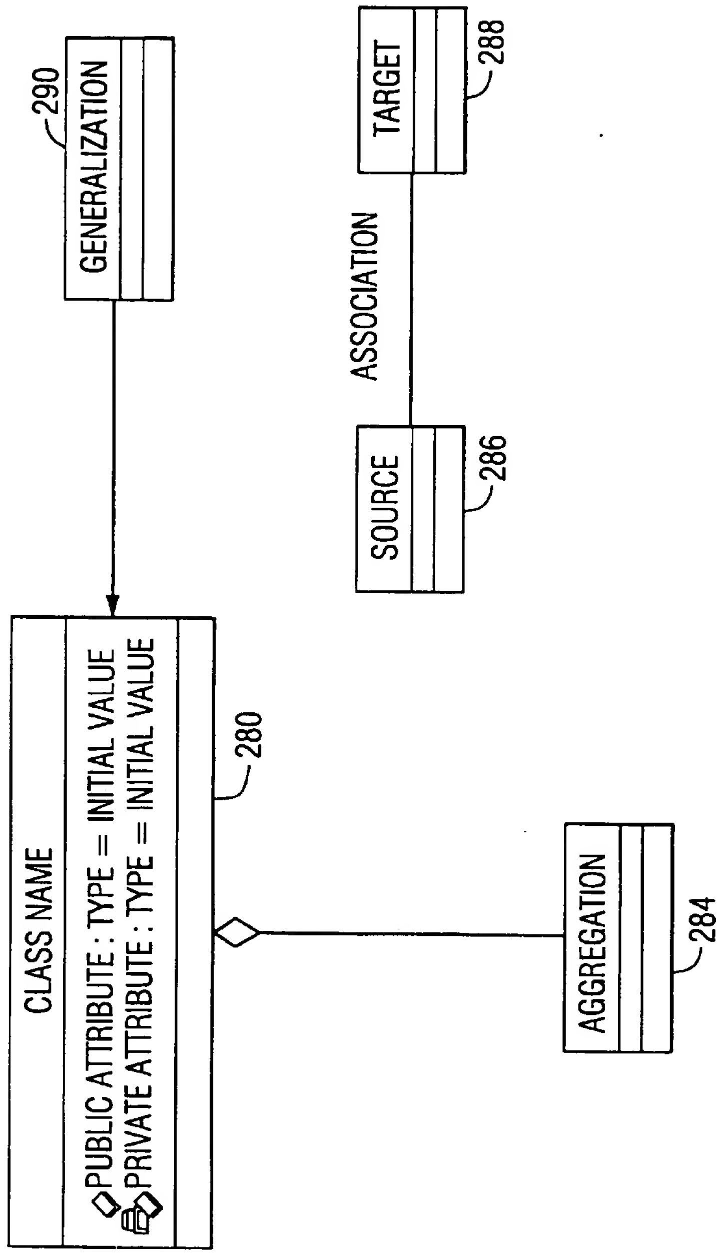
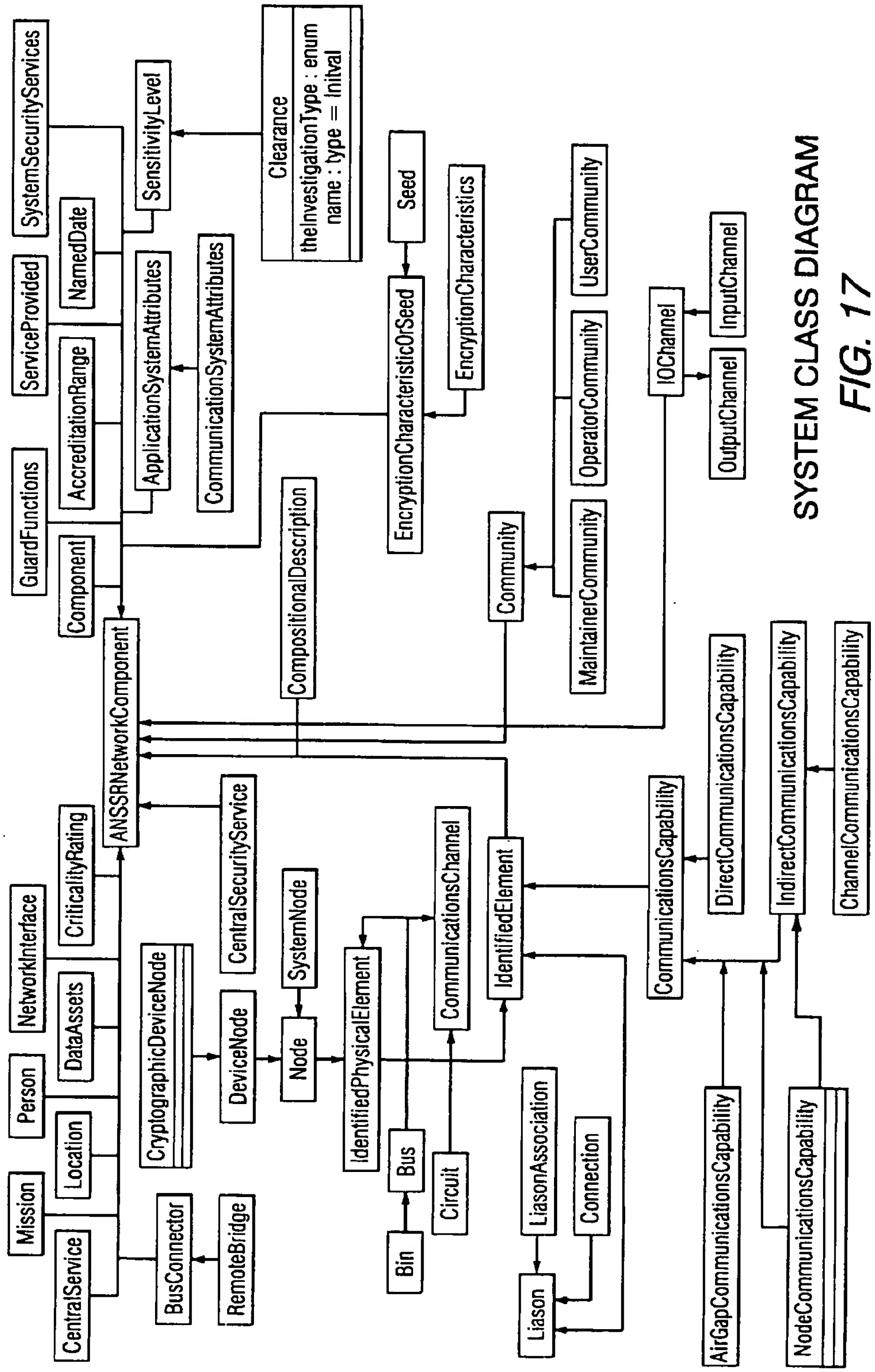

GENERALIZATION
290
TARGET
288
ASSOCIATION
SOURCE
286
CLASS NAME
PUBLIC ATTRIBUTE : TYPE = INITIAL VALUE
PRIVATE ATTRIBUTE : TYPE = INITIAL VALUE
280
AGGREGATION
284

FIG. 16



SYSTEM CLASS DIAGRAM

FIG. 17

SYSTEM AND METHOD FOR ASSESSING THE SECURITY POSTURE OF A NETWORK AND HAVING A GRAPHICAL USER INTERFACE

This invention was made with Government support under Contract No. F30602-96-C-0289 awarded by the United States Air Force. The Government has certain rights in this invention.

FIELD OF THE INVENTION

This invention relates to the field of networks, and more particularly, this invention relates to the field of assessing security vulnerabilities of networks.

BACKGROUND OF THE INVENTION

Information systems and computer network infrastructures currently under development are now being built with consideration for what constitutes an acceptable risk (or adequate protection). System assets, such as the hardware, software and system nodes of a computer network, must be protected to a degree consistent with their value. Additionally, these assets must be protected only until the assets lose their value. Any security features and system architecture should also provide sufficient protection over the life of the processed data. To assess whether or not any risk associated with a network is acceptable, a security engineer typically gathers all pertinent information, and then analyzes the risk associated with the network.

Risk analysis is a complex and time consuming process, which is necessary to determine the exposures within a network and their potential harm. As an example, when analyzing the security risks in a computer network, the security engineering typically follows the following steps:

1) Identify assets of the overall computing system.
2) Identify vulnerabilities of assets. This step typically requires imagination in order to predict what damage might occur to the assets and from what sources. The three basic goals of computer security are ensuring secrecy, integrity and availability. A vulnerability is any situation that could cause loss of one of those three qualities.
3) Predict likelihood of occurrence (exploitation), i.e., determining how often each exposure will be exploited. Likelihood of occurrence relates to the stringency of the existing controls and the likelihood that someone or something will evade the existing controls.
4) Compute any uncovered cost per year (expected annual loss) by determining the expected cost of each incident.
5) Survey applicable controls and their costs.
6) Project annual savings of control.

This last step of the analysis is a cost-benefit analysis, i.e., does it cost less to implement a control or to accept the expected cost of the loss? Risk analysis leads to a security plan, which identifies responsibility for certain actions to improve security.

Today, the rapid evolution of technology and proliferation of computers with increased power mandate the use of commercial-off-the-shelf (COTS) hardware and software components for cost effective solutions. This strong dependence on COTS implies that commercial grade security mechanisms are sufficient for most applications. Security architectures, therefore, must be structured to build operational, mission-critical computer systems with relatively weak COTS components. Higher assurance components can be placed at community or information boundaries, forming an enclave-based security architecture that implements a defense-in-depth approach to information assurance.

There are some design tools, i.e., software programs, available to the system architect to assist in maximizing the available protection mechanisms while remaining within the development budget. Current generation risk analysis tools usually are single vendor solutions that address a particular aspect or aspects of risk. These tools tend to fall into one of three categories:

1) Tools that work from documented vulnerability databases and possibly repair known vulnerabilities. Tools of this type are vendor-dependent for database updates, either through new product versions or by a subscription service. Examples from this category include ISS' Internet Scanner, Network Associates, Inc.'s CyberCop and Harris' STAT.
2) Monolithic tools that use various parameters to calculate a risk indicator. These tools are difficult to maintain and hard to keep current with the rapidly evolving threat and technology environment. An example of this tool category is Los Alamos Vulnerability Assessment (LAVA) tool.
3) Tools that examine a particular aspect of the system, such as the operating system or database management system, but ignore the other system components. SATAN, for example, analyzes operating system vulnerabilities, but ignores infrastructure components such as routers.

The use of multiple tools from a variety of vendors for a single computer network analysis is a labor-intensive task. Typically, a security engineer will have to enter a description or representation of the system (network) multiple times in multiple formats. The security engineer then must manually analyze, consolidate and merge the resulting outputs from these multiple tools into a single report of a network's security posture. Afterwards, the security engineer can complete the risk analysis (calculating expected annual loss, surveying controls, etc.), and then repeat the process to analyze alternatives among security risks, system performance, mission functionality and the development budget.

Also, none of these tools use an aggregate "snapshot" approach to the system with a "drill down" or layered approach to facilitate how one addresses risk at various layers (network, platform, database, etc.) of the system. These tools provide little assistance to system designers when analyzing alternatives among security risk, system performance and mission functionality. Instead, a "risk solution" is provided that addresses the particular aspect of risk that a given tool was designed to calculate. To develop a comprehensive risk assessment, a security engineer would have to become proficient in the use of several tools and manually correlate the resulting outputs.

One aspect of successful risk analysis is a complete and accurate accumulation of data to generate system models used by the analysis tools. Many current risk analysis tools depend on surveys filled out by users, system operations personnel, and analysts to acquire the data for development of a system model used in the analysis. Alternatively, a tool can actively scan a computer network to test various vulnerabilities against system-components.

However, these methods have drawbacks. Textual or survey-based knowledge solicitation techniques are labor intensive and potentially tedious for the analyst. Many of the existing tools reuse the same information to analyze differ-

ent aspects of the system security. It would be more advantageous to use a centralized repository of modeling data, which could provide a basis for shared inputs among existing tools. This repository could be used to generate data sets for use by risk analysis tools, allowing multiple tools to be run against the same system without separate input activities, thus reducing the possibility of operator error. The use of multiple risk analysis reasoning engines, or backbends, would allow various aspects of the system to be analyzed without the cost of developing one tool to perform all types of analysis. Integration of the information and the resulting informed assessments available by applying multiple tools would produce a more robust and accurate picture of a system's security posture. These results can facilitate more informed system design decisions, providing a framework for alternative evaluation and comparison.

SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a data processing system and method for assessing the security vulnerability of a network without having to analyze the network multiple times.

A graphical user interface is contained on a computer screen and used for determining the vulnerability posture of a network. A system design window displays network icons of a network map that are representative of different network elements contained within a network. The respective network icons are linked together in an arrangement corresponding to how network elements are interconnected within the network. Selected portions of the network map turn a different color indicative of a vulnerability that has been established for that portion of the network after vulnerability posture of the network has been established.

In still another aspect of the present invention, respective network elements determine different color indicative of a vulnerable network element. A graphical user interface can also comprise a manager window for displaying properties of network elements. A data sensitivity box can have user selected items for selecting the sensitivity of network elements. The graphical user interface can also comprise a select node configuration edit box having a user selectable vulnerability profile for selecting a vulnerability profile of a network node. The icons can be linked together by arrows that turn a different color indicative of a vulnerable connection that exists between those work elements.

Link

In still another aspect of the present invention, a graphical user interface is contained on a computer screen and used for determining the vulnerability posture of a network. It includes a system design window for displaying icons of a network map that are representative of different network nodes contained within a network. The respective icons are linked together in an arrangement corresponding to how network nodes are interconnected within the network. A manager window an be included and respective properties of network nodes can be displayed and edited. The selected icons turn the color red indicative of a higher risk node, and selected icons turn yellow indicative of a less severe risk node after a vulnerability posture of the network has been established.

Icon

The manager window further comprises a node properties dialog box for editing the properties of network nodes for network design alternatives. A graphical user interface can also comprise a data sensitivity box having user selected items for selecting the sensitivity of network nodes. A select node configuration edit box can have a user selectable vulnerability profile for selecting a vulnerability of a respective node.

In still another aspect of the present invention, a vulnerability posture window can display user readable items indicative of vulnerable network elements. The user readable items can comprise a chart indicative of vulnerable network elements and can comprise a spreadsheet indicating the vulnerable network elements.

BRIEF DESCRIPTION OF THE DRAWINGS

Other objects, features and advantages of the present invention will become apparent from the detailed description of the invention which follows, when considered in light of the accompanying drawings in which:

FIG. 1 is a schematic block diagram of a network showing locations where frequent problems are found on networks.

FIG. 2 is another schematic block diagram of a network showing an identified vulnerability located by the system and method of the present invention.

FIG. 3 is another block diagram showing overall architecture of the system and method of the present invention and showing filters used in association with the network model database.

FIG. 4 is another schematic block diagram of the architecture of the present invention showing the fuzzy logic analysis.

FIG. 5 is another schematic block diagram showing high level architecture components of the data processing system and method of the present invention.

FIG. 6 is another high level schematic block diagram of the data processing system of the present invention.

FIG. 7 is an example of a graphical user interface that models the network as a map.

FIGS. 8A and 8B show open windows that provide data resolution in the establishment of the system object model database.

FIG. 9 is an example of a graphical user interface showing the network model.

FIG. 10 is a graphical user interface showing various reporting options for the security posture of the network.

FIG. 11 is a block diagram showing the basic processing components of the goal oriented fuzzy logic processing used in the data processing system and method of the present invention.

FIG. 12 is a schematic block diagram of the data fusion used in the data processing system and method of the present invention.

FIG. 13 is another schematic block diagram showing an example of gold-based fusion rules used in the data processing system and method of the present invention.

FIG. 14 is another block diagram showing basic processing steps and components used in the fuzzy logic processing of the data processing system and method of the present invention.

FIG. 15 is a block diagram showing basic components used in the fault tree analysis (DPLf) for evidence accumulation and fuzzy evidential reasoning rules.

FIG. 16 is a block diagram showing an object/class hierarchy.

FIG. 17 is a block diagram showing the system class diagram of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 1 illustrates an example of a conventional network 100 having internal servers 102 that connect to an external

router 104, communication network 105, and firewall 106. An internal router 108 is connected to the firewall 106, branch office 107, and connected to internal LAN network components 110 and a remote-access server 112 and remote user 114.

Using the example of FIG. 1, frequent problems found on networks include hosts, such as the internal servers 102, which run unnecessary services, for example, a denial of service and anonymous FTP or misconfigured web servers that could be an internal server, for example, CGI scripts, anonymous FTP and SMTP. The internal LAN's 110 could include unpatched, outdated, vulnerable or default configured software and firmware and weak passwords. LAN's could also include improperly exported file sharing services, such as NetWare file services and NetBIOS. The internal LAN 110 could also include misconfigured or unpatched windows NT servers and problems caused by a lack of comprehensive policies, procedures, standards and guidelines. A remote-access server 112 could have unsecured remote-access points and the external router 104 could have information leakage through services, such as SNMP, SMIP, finger, roosers, SYSTAT, NETSTAT, TELNET banners, Windows NT TCP 139 SMB (server message block), and zone transfers to non-named server hosts. It could also have inadequate logging, monitoring and detecting capabilities. The branch office 107 could have a misappropriated trust relationship such as RLOGIN, RSH, or REXEC. The firewall 106 could be misconfigured or have a misconfigured router access control list.

Although these network problems are only an example of common problems found on networks 100, there are many other problems that could occur, as is well known to those skilled in the art.

The present invention is advantageous because the system and method of the present invention allows the vulnerabilities of a network system to be identified. The software of the data processing system and method can be located on a user terminal 120, as shown in FIG. 2, showing an identified vulnerability of a node 112 connected in the internal LAN 110. For purposes of description, the data processing system and method of the present invention can be referred to as a Network Vulnerability Tool (NVT), i.e., a tool a user uses to determine network vulnerabilities and risks.

The data processing system forming the NVT of the present invention can be loaded on a Pentium PC platform running Windows NT. This type of platform can provide a low cost solution and support a large variety of assessment tools, also commonly referred to as network vulnerability assessment or risk analysis programs throughout this description. These network vulnerability analysis programs typically are the standard COTS/GOTS programs known by security engineers, and include HP Open View, which allows network automatic discovery or manual network modeling; ANSSR (Analysis of Network System Security Risks) as manufactured by Miter Corporation, a GOTS network system analysis tool, which allows passive data gathering and single occurrence of loss. NSA's risk assessment methodology known as RAM (risk assessment model) can also be used and is implemented in the DPL-f decision support programming language. RAM also allows passive data gathering for event tree logic, prioritizes the task list, and allows a mathematical model with multiple risks/services. It is event based over time.

DPL (decision programming language) is a decision support software package that facilitates the modeling of complex decisions. It allows a user to incorporate uncertainty and flexibility into a decision process. DPL provides a graphical interface for building a model, and performs analyses on the model. DPL-f contains the functionality built into DPL and provides a graphic interface for fault tree construction. This feature allows the modeler to create fault trees and incorporate them into DPL models. DPL-f also contains unique analytic tools. These tools include the ability to calculate explicitly the probability of any event in the tree and perform fault tree-specific types of sensitivity analysis. DPL-f provides an interface for incorporating time series into a model. This allows a modeler to account for devaluation, capital growth or other time-bearing quantities without changing the structure of the model. DPL-f provides RAM with additional capabilities for rapid fault tree construction, libraries of embedded fault trees, an expert opinion generation system, enumeration and ordering of cut sets and a graphical portrayal of risk over time.

The ISS Internet scanner as developed by Internet Security Systems Corporation (ISS) allows active data gathering and scans a network for hosts, servers, firewalls and routers and assesses security and policy compliance with networks, operating systems and software applications. It allows a snapshot in time and a computer network compliance report. These programs are disparate network vulnerability analysis programs that the NVT of the present invention allows for integration.

The NVT of the present invention is based on a knowledge solicitation framework, which incorporates a graphical description of a network topology. This topology is used to capture network attributes and analyzed subsequently for security vulnerabilities. Graphical user interface is also used to improve accuracy of the network model.

In accordance with the present invention, the system and method of the NVT automatically maps an existing network and can display the existing network as a model on a graphical user interface, such as shown in FIG. 7. For example, HP Open View could graphically depict a network topology. Once the software has been given the IP address of a default router for the network, the NVT of the present invention can use Open View and search for computers and other devices attached to the network. NVT performs an active search, pinging possible IP addresses on the network, and adding whatever response information it receives to its network map. NVT also provides a manual method to draw a proposed network with the graphical user interface, as illustrated, to support drag and drop. A system architecture can be defined, including security critical information for alternative designs or node editing to provide additional details as required to provide complete logical network planning. A user can also represent an entire network on a map by using a sub-network icon.

When a network system description has been completed, the NVT of the present invention represents and stores the description in an object/class hierarchy, as shown as an example in FIGS. 16 and 17, as will be explained below. A single topological system object model supports the information data needs of the disparate network vulnerability analysis programs (tools). Fuzzy logic processing of the results allows correlation of the results from the programs into a cohesive vulnerability/risk assessment to obtain a vulnerability posture of the network, as shown in the graphical user interface at FIG. 10. The single representation of the system simplifies the use of multiple tools and eliminates redundant data entry. It also provides a foundation for addressing the problem of incomplete data for a given vulnerability assessment tool and future knowledge negotiation capabilities.

FIG. 3 illustrates at 130 an example of the overall network visualization tool (NVT), data processing system of the present invention, where three network vulnerability analysis programs (tools) are illustrated as ANTSSR 132, ISS Internet scanner 134, and RAM 136. The system and method of the present invention creates a system object model database (Network Model DB) 138 that represents a network and supports the information data requirements of the network vulnerability analysis programs. The system object model database 138 represents a single representation of the assessed system or design, and addresses the need for a single internal representation of a network to provide data to the network vulnerability analysis programs.

This model 138 uses object oriented (OO) methodology to provide an extensible set of components in a class hierarchy that can be combined to represent a network. The class hierarchy provides a means of defining components with shared common traits, while retaining the specifics that distinguished it from other components. In addition to an implicit hierarchical relationship, object oriented techniques provide a containment mechanism in which an object can contain a reference to any object, including itself. This provides a flexible mechanism for representing-any physical or logical entity. Also, object oriented representation lends itself to ready modification and extension and is ideal for an information assurance arena where changes and new technologies arise daily.

As shown in FIG. 3, filters 140 are associated with each of the network vulnerability analysis programs 132, 134, 136 and allow only that data required by a respective network vulnerability programs to be exported to the tool (program). The filters are a C++ base class that provide a set of virtual methods to allow data movement between the NVT system and a program. The filter also provides a means for the NVT to control execution of the tool and complete data needed by a tool. NVT views each tool as a filter, calling the appropriate method within the filter to perform the desired task, including initializing, running, importing data and exporting data. Each tool can have a concrete filter subclass and provide the means to define each method specifically for the tool, while still providing the generic and well-defined programming interface (API) to NVT. This allows all tools to be treated the same within NVT, allowing the addition and removal of tools without changing any of the existing NVT codes.

Establishing communication between DPL-f and NVT using the filter technology is straightforward. A DPL-f filter is tasked with the specifics of building and populating fault trees. As an analysis tool, a default tree can represent a node in a network as developed and provide a probability value for events such as denial of service, loss of data and data compromise. Actually, DPL-f can be used as a final result tool.

The network is then analyzed with each network vulnerability analysis program to produce data results from each program. The data results are correlated to determine a security posture of the network. Network validation can occur through the fuzzy logic processing of the invention, as will be explained below, and the system GUI can have input to a user display.

An overview of the network is created as a model 142 by an automatic network discovery or manual entry 144, such as through HP Open View, and an appropriate filter 146 allows the system GUI 148 to display the network model as shown in FIG. 7 via an appropriate data input 150 to a user display 152. It is also possible to have a risk GUI 154 to assess visually the risk vulnerability, a log 156 of the risk/vulnerability report, a risk assessment 158 as part of the GUI 148, all through the network validation 160, using a plug-in or fuzzy rule set as will be described in greater detail below. Any incomplete data resolution 161 can also be handled.

FIG. 4 illustrates a high level block diagram similar to FIG. 3, showing the system object model database 138 that can be established and work in conjunction with an integrated application programming interface 162 to allow importing of data into the various tools 164, as illustrated as a model tool, discovery tool and information analysis tools that result in the overall system results database 166. An application programming interface 168 and a graphical user interface 170 work in conjunction with model database 138. An evaluation/assessment manager 172 (manager) works in conjunction with an application programming interface (API) 174 and graphical user interface (GUI) 176 to correlate data results with fuzzy logic processing, indicated by dotted lines 178, including expert correlation 180 and fuzzy inferences and evidential reason 182 to produce vulnerability results 184 and a graphical user interface (GUI) 186 for the correlated results. Although FIG. 4 represents a high level model showing an example of different components, it is only one example of one type-of high level components that could be used with the NVT system and method of the present invention.

FIGS. 5 and 6 illustrate other examples of high level models showing basic components and processing steps of the data sources 200 (FIG. 5), together with the system picture 202, a per tool analysis 204, a multi-tool analysis 206, the tool-to-expert analysis 208, and report media 210. The tool-to-expert analysis 208 could include the DPL-f 208*a* as part of the fuzzy logic processing in a data fact base, and use with CERT notes 208*b* and an expert system 208*c* for expert correlation. Reports can be generated, including output as icons on a graphical user interface, text, an EXCEL spreadsheet, Access and Configuration, as known to those skilled in the art. FIG. 6 also illustrates another high level model similar to FIG. 5, where the tools used to form a complete system object model and fuzzy logic process could include the individual tool processing and the multi-tool correlation.

FIGS. 7–10 illustrate in greater detail a graphical user interface 220 that can be contained on a computer screen and used for interacting with the NVT and determining the vulnerability posture of a network. As illustrated, the graphical user interface 220 is a standard type of Windows™ interface. A system design window 222 permits the display of network icons 224 forming a network map that is representative of the relationship among different network elements and nodes contained within a network. Respective network icons 224 are linked together in an arrangement corresponding to how the network elements nodes are interconnected within the network. As shown in FIG. 7, the network elements can be linked together via connection lines 226, showing the interconnection that exists among actual network elements and nodes. The system design window 222 shows on the left side an internetwork view 230 with two nodes and a network view 232 on the right hand side of the window to illustrate a map of the network model. A manager window 234 is opened and displays properties of network elements.

A select data sensitivity pop up window (box) 240 is user selectable through the menu options for selected network elements (FIG. 8A), and has user selected items for selecting the sensitivity of network elements. The sensitivity for data

on any node (node **1** in the example shown in FIG. **8A**) can be selected for unclassified, sensitive, confidential, secret, restricted secret or top secret with appropriate Okay, Random and Default buttons.

A select node configuration edit pop up window (box) **250** is shown in FIG. **8B** and can have user selectable vulnerability profiles for selecting a vulnerability profile of a network element or node. FIG. **9** also shows the network model diagram with the central hub and the interconnected nodes. It is possible that a user can edit the manager window **234** entries, which also allows the network discovery to occur through appropriate selection of buttons. Naturally, network icons can be selected and moved as necessary for editing and design alternatives.

After the security posture has been established through the system, icons representative of high risk network elements can turn colors, such as red, the hub **252**. Other selected icons could turn yellow, indicative of a less severe risk node, such as the HP4 node **254** shown in FIGS. **7** and **9**. It is possible that shaded areas around the node or portions of the network could be colored red or yellow indicative of higher risk vulnerability. It is also possible that the connection line could turn red or yellow to indicate a poor connection between elements.

FIG. **10** illustrates a vulnerability posture window **270** for displaying user readable icons indicative of vulnerable network elements and icons. The overall system model is shown as part of an open system design window. However, a spreadsheet **272** is illustrated and a NVT risk assessment chart **274** having slider bars for risk assessment. A risk analysis window **276** showing the top five risk analysis elements is also illustrated.

FIG. **16** shows in greater detail a class hierarchy with the Class Names **280** as public attributes and private attributes, the Aggregation **282** and Association **284** of Source **286** and Target **288** with Generalizations **290**. FIG. **17** illustrates an example of a system class diagram with various components identified in the blocks. Naturally, FIG. **17** is only a system class diagram as is known to those skilled in the art and is an example of what can be used for the system and method of the present invention.

Referring now in greater detail to FIGS. **11–15**, the goal oriented fuzzy logic decision making is illustrated. As shown in FIG. **11**, the system model database **138** and results **300** from the respective network vulnerability analysis programs are combined together using an applications programming interface and expert correlation to form a data fact base **302** through data fuzzification. Goal oriented fuzzy logic decision rules operate through fuzzy inference network rules **304** and fuzzy evidential reasoning rules **306** to determine the security posture of a network based on predetermined goals **308**.

The fuzzy logic processing of the present invention uses data fusion, evidential reasoning and inference network techniques. As known to those skilled in the art, evidential reasoning is a technique in which facts are gathered that support and refute a given hypothesis. The result is the proof or rejection of the hypothesis with a certain degree of confidence. The fuzzy logic processing of the present invention uses evidential reasoning to accumulate evident from the system and tool findings for each criteria, thereby merging the system assessment data into a single point of reference, the conformance of the system to a particular criteria. By suppling a set of criteria for fusion, the system constrains the fusion problem and reduces the search base. Evidential reasoning has previously been used to perform level-one multi-sensor data fusion, and is a common global reasoning technique in fuzzy expert systems, such as the type of system known to those skilled in the art as fuzzyCLIPS, developed by NASA. The result is a set of fuzzy evidential rules whose purpose is to accumulate evidence for a given set of requirements. This resolves potentially conflicting, ambiguous and redundant data from expert correlation and draws conclusions with available data, even if it is incomplete.

The accuracy of the result is contingent upon the quantity and quality of the data available and it may be necessary to perform additional refinement on the available data prior to the application of fuzzy logic processing, while also maintaining the probabilistic nature of the data. This refinement uses inference networks and provides a method of reasoning about probability using heuristics, thereby removing the need for extensive a priori knowledge. The relation between the goals and potential security metrics encourages cross fertilization. As known to those skilled in the art, the fuzzyCLIPS uses fuzzy facts, which can assume any value between 0 and 1. The result can be viewed as a two dimensional plot of a continuous function bounded vertically by 0 and 1.

Data fusion is used with the system object database, data results data fact base. Intelligence data fusion is a multilevel, multi-disciplinary-based information process to yield the integration of information from multiple intelligence sources (and perhaps multiple intelligence disciplines) to produce specific and comprehensive, unified data about an entity (its situation, capabilities, and the threat it imposes). Data fusion provides information based on the available inputs. The intelligence data fusion process is typically partitioned into four levels, described in Table 1 below.

TABLE 1

THE LEVELS AND PURPOSES OF THE INTELLIGENCE DATA FUSION PROCESS

Data Fusion Level	Description
1 Object Refinement	• Transforms data into consistent frame of reference • Refines and extends, in time, estimates of object position, kinematics or attributes • Assigns data to objects to allow application of estimation process • Refines the estimation of object identity
2 Situation Refinement	• Develops description of current relationships among objects and events in the context of the environment • A symbolic, reasoning process by which distributions of fixed and tracked entities and events and activities are associated with environmental and performance data in the context of an operational problem
3 Threat Refinement	• Projects the current "situation" into the future and draws inferences about threats, vulnerabilities and opportunities for operations
4 Process Refinement	• Monitors process performance to provide information for real-time control and long-term improvement • Identifies what information is needed to improve the multi-level fusion product • Determines the source specific data requirements to collect required information • Allocates and directs the sources to achieve mission goals

As noted before, NVT combines multiple types of data, from multiple sources, with other contextual information to form an integrated view of a networked system's security posture. NVT provides a user with a simple expression of the vulnerability posture of a given system or system design, and enables them to perform "what if" analysis for functionality, performance, and countermeasure trades, for the purpose of refining and improving the system or system design.

In computer security engineering, sensors are the various vulnerability assessment and risk analysis tools, along with the GUI to gather information, as needed, from the user. The resulting outputs from these tools take the form of both qualitative and quantitative data, in a variety of formats from different vendors. For computer security engineering, the objects of interest are the nodes in a network (computing system), i.e. the assets, including hardware, software and data. The situation of interest is an assessment of the weaknesses in the security system of a computer network segment that might be exploited to cause harm or loss of secrecy, integrity or availability.

Assessing the risk faced by a computing system involves an assessment of the threats faced, their likelihood of occurrence (exploitation), and the expected cost of the loss (or harm). Finally, the network (computing system) can be refined based on the results of cost-benefits analysis. This requires information on protective measures (controls or countermeasures) appropriate for particular vulnerabilities and their costs. The cost-benefit analysis seeks to determine if it costs less to use a control or countermeasure, or accept the expected cost of the loss. This leads to the development of a security plan to improve security of a computer network system.

Table 2 contains an example of a first partitioning of this data fusion process for computer security engineering that could be used with the present invention, with four processing levels, corresponding to the four levels found in Table 1. As illustrated in FIG. **12**, inputs to this process would consist of the object model database **138**, results from individual tools **132**, **134**, **136**, and other contextual information. The different data fusion levels 1–4 are indicated generally at **320**, **322**, **324** and **326**.

TABLE 2

INITIAL PROCESSING LEVELS OF DATA FUSION FOR COMPUTER SECURITY RISK ANALYSIS

	Data Fusion Levels	Description
1	Node Data Refinement	Transforms data into consistent frame of reference • Refinement of data at the network node-level (the objects for computer security data fusion) • Data from multiple tools - correlated (assigned to appropriate nodes) and possibly combined at each node • Refines the estimation of object identity - network node (workstation) is a system-of-systems, consisting of an OS, critical applications, a database and data • Vulnerability analysis at this level does not yet constitute situation assessment
2	Network Segment Refinement	• Refinement of the situation at the network segment-level (system-of-systems level) • Develops description of current relationships among objects (nodes) in the context of the environment (a network segment)

TABLE 2-continued

INITIAL PROCESSING LEVELS OF DATA FUSION FOR COMPUTER SECURITY RISK ANALYSIS

	Data Fusion Levels	Description
		• A symbolic, reasoning process by which information about entities (nodes, network segments) and environment are associated with evidence about computer security goals, requirements • Combining tool results at the network segment-level • The situation of interest is the assessment of the network segment's vulnerabilities or exposures
3	Risk Refinement	• Refinement of the exposures and their potential for harm (risk) within a computing system • Projects the current "situation" (state of the computer network system) into the future and draws inferences about threats, vulnerabilities and opportunities for operations • Based on vulnerabilities, concerns, context, cost, threats • Refinement of a system design with the identification of controls that reduce one or more vulnerabilities • Based on countermeasures, components, cost • Identifies what information is needed to improve the multi-level fusion product • Facilitate long-term improvement of the system

While the data fusion used in the present invention provides a conceptual framework for addressing the problem of merging results from multiple vulnerability assessment and risk analysis tools, expert systems, inference networks and evidential reasoning are used to implement the fusion concepts and merge tool results. The flexibility of fuzzy decision technology, in particular, fuzzy expert systems, offers the means to address these problems. A primary benefit of a fuzzy expert system is its ability to use and assimilate knowledge from multiple sources.

Fuzzy logic provides the technique for representing and inferring from knowledge that is imprecise, uncertain or unreliable. Similar to traditional expert systems, a fuzzy expert system can represent knowledge in the form of a system of IF/THEN rules in which the antecedents, consequent, or both are fuzzy rather than crisp. Fuzzy logic is used to determine how well fuzzy facts match the rules, and to what degree this match affects the rule's conclusion.

In accordance with the present invention, an inference network is a hierarchy of heuristic rules that can propagate probabilities without requiring extensive knowledge of a priori probabilities (e.g. Bayesian networks). The heuristic rules can be developed using expert knowledge on how the probabilities propagate, allowing conclusions to be drawn with limited knowledge of a priori probabilities. This results in low-level discrete probabilities being accurately reflected in higher-level conclusions. Probabilities of low-level events (such as probability of password compromise based on lifetime) need to be part of any conclusions drawn on higher-level events (vulnerability of password).

Initial studies of NVT uses accumulation of evidence to modify a fuzzy-fact and represent the change in state required by the current system. This state change fuzzy-fact is then used to modify the system and the new state is fed

back into the change of state rules in an endless cycle, using global contribution. FuzzyCLIPS allows the definition of fuzzy-fact types, but only one fact of each type will ever exist. Therefore every rule that manipulates that fact type actually modifies a single fact, leading to accumulation of evidence.

Global contribution and accumulation of evidence have lead to a FuzzyCLIPS methodology that defines fuzzy-facts representing different vulnerability states. These facts will use global contribution and accumulation of evidence to acquire final values reflecting the tested system's vulnerability, i.e., evidential reasoning. This method reflects the well-defined use of fuzzy logic control systems, limiting the execution to a finite number of cycles instead of allowing it to run continuously. FuzzyFusion™ has been developed by Harris Corporation of Melbourne, Fla., and will use this methodology to accumulate evidence from rules based on knowledge from network security experts. In particular, FuzzyFusion™ will employ evidential reasoning as a technique in which facts are gathered supporting and refuting a given hypothesis. The result is the proof or rejection of the hypothesis with a certain degree of confidence.

Initial knowledge extraction has resulted in the use of security requirements to accumulate evidence, i.e. how well does a system meet the requirements. This demonstrates a strong correlation between the methods of verifying a database (e.g. AFCERTS) and verifying security requirements, leading to using the database and requirements as global contribution facts to accumulate evidence, illustrated in FIG. 13. This also shows how varying the granularity of the goals directly impacts the granularity of the assessment, i.e. the assessment will only be as detailed as the goals. The accumulation of evidence is being viewed as a goal orientated approach to obtaining the results while maintaining the use of a forward inference technique, and for now will be phrased as "Goal-based Fusion".

One example of how fuzzy logic can be applied with merging tool results in computer security uses the combination of results from ANSSR and ISS Internet Scanner, two of the tools currently used within one aspect of NVT. The outputs of the tools are both quantitative (ANSSR) and qualitative (Internet Scanner). Fuzzy logic allows the system to represent both data types within the same system. Then an initial hypothesis is formulated, and fuzzy logic is used to gather evidence to contradict or support the hypothesis.

For this example, an initial hypothesis could be that auditing is invalid in an existing network system. The system user then exercises the ANSSR and ISS Internet Scanner tools. If ANSSR supplies a number 90 (out of 100), that auditing is sufficient. Fuzzy logic allows NVT to account for this as strong refuting evidence for the initial hypothesis that auditing is invalid. If Internet Scanner supplies the qualitative data that User Access is not audited, fuzzy logic accounts for this as supporting evidence, which is combined with the evidence from ANSSR. When the tools are finished, the contributing evidence for auditing is represented as a single fuzzy fact that provides a measure of how well auditing is implemented.

FuzzyFusion™ as developed by Harris Corporation of Melbourne, Fla. is a means to consolidate and merge the results of vulnerability assessment and risk analysis tools, employed within the NVT into a unified report. In particular, FuzzyFusion™ is developed to implement Levels 1 and 2 fusion. FuzzyFusion™ is accomplished through the use of a fuzzy expert system (Goal-Oriented Fuzzy Logic Decision Rules) using FuzzyCLIPS, which combines the outputs of the various tools, user concerns about system risks and vulnerabilities, and expert understanding of the results of each tool and how these fit into a larger information system security picture. Thus, NVT users obtain a simple expression of the security posture of a given computing system, or system design, and can perform "what if" analysis for functionality, performance, and countermeasure trades.

FIG. 14 illustrates the NVT FuzzyFusion™ component architecture for implementing the first two levels of data fusion for computer security engineering. As the figure illustrates, the task of modeling security expertise is partitioned into discrete tasks. Separation of Expert Correlation (Data Framework Merge Rules), Fuzzy Inference Network Rules, and Fuzzy Evidential Reasoning Rules addresses the problems of brittle expert systems and computational explosion. It also segregates low-level data correlation and fusion from the resolution of ambiguous/conflicting data and the merging of results into one picture. This should result in fuzzy expert systems that are easier to maintain than one large comprehensive system. Elements of this architecture are described below.

Data Fuzzification **310** converts the results from the individual vulnerability assessment and risk analysis tools **132, 134, 136** into fuzzy-facts, and stores those along with the Common System Model (CSM), i.e., system object model database **138**, into the (FuzzyCLIPS) Fact-Base **302**. Individual tool results (after fuzzification) and the CSM **138** are exported for Expert Correlation processing **3310** (Data Framework Merge Rules) to resolve system information and integrate tool output based on security expertise. Expert opinion can be used to determine the specific fuzzy values attributed to the low-level events.

The Expert Correlation (Data Framework Merge Rules) **330** is a collection of fuzzy expert rules to perform node-level data refinement (Level-1) or network-segment refinement (Level-2). These rules correlate and consolidate the (fuzzified) outputs from the vulnerability assessment and risk analysis tools, using expertise from security engineers. These rules leverage extensive experience in security assessment to resolve low-level systems data and tool results. These rules resolve system information and integrate tool output. Expert Correlation Rule processing **330** can also transform low-level data from the CSM and tool results into high level conclusions. For example,

IF auditing is on with these flags, AND the audit data is not backed up, THEN auditing is unreliable.

Working from fuzzy-facts in the Fact-Base **302**, a set of Level-1 fusion rules can consolidate the vulnerabilities for each node, resulting in a vulnerability rating for each node in the network. This rating can be imported back to NVT for display. Similarly, a set of Level-2 fusion rules can consolidate the vulnerabilities for each network segment, resulting in a vulnerability rating for each network segment. This can again be imported back for display.

The data is then subject to Fuzzy Inference Network Rules processing **304**. It may be necessary to perform additional refinement on the available data prior to the application of Fuzzy Evidential Reasoning Rules **304**, while maintaining the probabilistic nature of the data. This refinement will use inference networks, as known to those skilled in the art, which provides a method of reasoning about probability using hueristics, thereby removing the need for extensive a priori knowledge.

Fuzzy Evidential Reasoning Rules **306** are a collection of fuzzy expert rules to merge individual tool results into a higher level assessment, from a systems-level perspective,

of a network's security posture. These rules provide a mechanism for merging the CSM, tool results and the results from the Expert Correlation (Data Framework Merge Rules) **330** into a unified report. This also removes the necessity of dealing with incomplete and conflicting data from the forward-chaining expert system used in Expert Correlation.

Evidential reasoning use a technique in which facts are gathered to support and refute a given hypothesis. The result is the proof or rejection of the hypothesis with a certain degree of confidence. FuzzyFusion™ uses evidential reasoning to accumulate evidence from the Common System Model and tool findings for each criterion, thereby merging the computer network system assessment data into a single point of reference, the conformance of the system to particular criteria. By supplying a set of criteria for fusion, NVT constrains the fusion problem and reduces the search space, referred to earlier as goal-based fusion. The result will be a set of fuzzy evidential rules whose sole purpose is to accumulate evidence for a given set of requirements. This resolves the potentially conflicting, ambiguous and redundant data from Expert Correlation (Data Framework Merge Rules) **330**, and draws conclusions with the available data, even if it is incomplete. Obviously, the accuracy of the result is contingent upon the quantity and quality of the data available.

As noted before, the fuzzy logic processing is goal oriented. Goals for Evidence Accumulation processing **350** may be derived from a Security Requirements Database **352**, a Computer Security Metrics Database **354**, or a Vulnerability Database **356**, such as, a database composed of AFCERTs. Bounding fusion to pre-defined goals limits computation times. FuzzyFusion™ goals provide mechanism to obtain IA metrics.

The FuzzyFusion™ process has a number of advantages over traditional approaches. Crisp expert systems would require extremely large knowledge bases to encompass the necessary data and, yet, would still have a problem with incomplete data and conflicting results. Bayesian and probability networks require extensive and often unavailable a priori knowledge of probabilities. Algorithmic solutions do not fit the probabilistic and heuristic nature of the security problem.

Rete-based expert systems such as FuzzyCLIPS suffer from a geometric increase in execution time based on the number of rules and facts present in the system. This leads to breaking the analysis into subnetworks. FuzzyFusion™ will add subnetwork and scaling capabilities. The nodes for each subnetwork will be evaluated as a group, and then groups of subnetworks will be evaluated. Grouping the rules for each type of analysis into different modules will reduce the size of the Rete-network. In addition to decreasing execution time, this will also introduce a scalable method of analyzing networks that maps to the network model used by NVT.

As shown in FIG. **15**, the other possible data spaces could include a threat knowledge database **360**, cost database **362** as part of Level 3 fusion and a counter measure knowledge base, component database and cost database as part of Level 4 fusion.

This application is related to copending patent applications entitled, "SYSTEM AND METHOD FOR ASSESSING THE SECURITY POSTURE OF A NETWORK," and "SYSTEM AND METHOD FOR ASSESSING THE SECURITY POSTURE OF A NETWORK USING GOAL ORIENTED FUZZY LOGIC DECISION RULES," which are filed on the same date and by the same assignee and inventors, the disclosures which are hereby incorporated by reference.

Many modifications and other embodiments of the invention will come to the mind of one skilled in the art having the benefit of the teachings presented in the foregoing descriptions and the associated drawings. Therefore, it is to be understood that the invention is not to be limited to the specific embodiments disclosed, and that the modifications and embodiments are intended to be included within the scope of the dependent claims.

That which is claimed is:

1. A graphical user interface contained on a computer screen and used for determining the security posture of a network comprising:

a system design window for displaying network icons of a network map that are representative of different network elements contained within a network, wherein respective network icons are linked together in an arrangement corresponding to how network elements are interconnected within the network;

wherein selected portions of the network map turn a different color indicative of a vulnerability that has been established for that portion of the network after a security posture of the network has been established by correlating a system object model database that supports information data requirements of disparate network vulnerability analysis programs with any data results obtained from the programs.

2. A graphical user interface according to claim **1**, wherein respective network elements turn a different color indicative of a vulnerable network node.

3. A graphical user interface according to claim **1**, and further comprising a manager window for displaying properties of network elements.

4. A graphical user interface according to claim **1**, wherein icons are linked together by arrows that turn a different color indicative of a vulnerable connection that exists between network elements.

5. A graphical user interface contained on a computer screen and used for determining the security posture of a network comprising:

a system design window for displaying icons of a network map that are representative of different network nodes contained within a network, wherein respective icons are linked together in an arrangement corresponding to how network nodes are interconnected within the network;

a manager window on which respective properties of network nodes are displayed and edited;

wherein selected icons turn the color red indicative of a higher risk node and selected icons turn yellow indicative of a less severe risk node after a vulnerability posture of the network has been established by correlating a system object model database that supports information data. requirements of disparate network vulnerability analysis programs with any data results obtained from the programs.

6. A graphical user interface according to claim **5**, wherein said manager window further comprises a node properties display box for editing the properties of network nodes for network design alternatives.

7. A graphical user interface according to claim **5**, and further comprising a data sensitivity box having user selected items for selecting the sensitivity of network nodes.

8. A graphical user interface according to claim **5**, and further comprising a select node configuration edit box having a user selectable vulnerability profile for selecting a vulnerability of a respective node.

9. A graphical user interface contained on a computer screen and used for determining the security posture of a network comprising:

a system design window for displaying network icons of a network map that are representative of different network elements contained within a network, wherein respective network icons are linked together in an arrangement corresponding to how network elements are interconnected within the network, wherein selected portions of the network map turn a different color indicative of a vulnerability that has been established for that portion of the network after a security posture of the network has been established by correlating a system object model database that supports information data requirements of disparate network vulnerability analysis programs with any data results obtained from the programs; and

a vulnerability posture window for displaying user readable items indicative of vulnerable network elements.

10. A graphical user interface according to claim 9, wherein said user readable items comprise a chart indicative of vulnerable network elements.

11. A graphical user interface according to claim 9, wherein said user readable items comprise a spreadsheet indicating the vulnerable network elements.

12. A graphical user interface according to claim 9, wherein respective network elements represented by icons turn a different color indicative of a vulnerable network node.

13. A graphical user interface according to claim 9, and further comprising a manager window for displaying properties of network elements.

14. A graphical user interface according to claim 9, and further comprising a data sensitivity box having user selected items for selecting the sensitivity of network elements.

15. A graphical user interface according to claim 9, and further comprising a select node configuration edit box having a user selectable vulnerability profile for a network node.

16. A graphical user interface according to claim 9, wherein icons are linked together by arrows that turn a different color indicative of a vulnerable connection that exists between network elements.

17. A graphical user interface contained on a computer screen and used for determining the security posture of a network comprising:

a system design window for displaying icons of a network map that are representative of different network nodes contained within a network, wherein respective icons are linked together in an arrangement corresponding to how the network nodes are interconnected within the network;

a manager window on which respective properties of network nodes are displayed and edited;

wherein selected icons turn the color red indicative of a higher risk node and selected icons turn yellow indicative of a less severe risk node after a security posture of the network has been established by correlating a system object model database that supports information data requirements of disparate network vulnerability analysis programs, with any data results obtained from the programs; and

a vulnerability posture window for displaying user readable items indicative of vulnerable network icons.

18. A graphical user interface according to claim 17, wherein said user readable items comprise a chart indicative of vulnerable network nodes.

19. A graphical user interface according to claim 17, wherein said user readable items comprise a spreadsheet indicating the vulnerable network nodes.

20. A graphical user interface according to claim 17, wherein said manager window further comprises a node properties display box for editing the properties of network nodes for network design alternatives.

21. A graphical user interface according to claim 17, and further comprising a data sensitivity box having user selected items for selecting the sensitivity of data respective nodes.

22. A graphical user interface according to claim 17, and further comprising a select node configuration edit box having a user selectable vulnerability profile for a respective node.

23. A graphical user interface contained on a computer screen and used for determining the security posture of a network comprising:

a system design window for displaying network icons of a network map that are representative of different network elements contained within a network, wherein respective network icons are linked together in an arrangement corresponding to how network elements are interconnected within the network and a data sensitivity box having user selected items for selecting the sensitivity of network elements;

wherein selected portions of the network map turn a different color indicative of a vulnerability that has been established for that portion of the network after a security posture of the network has been established.

24. A graphical user interface contained on a computer screen and used for determining the security posture of a network comprising:

a system design window for displaying network icons of a network map that are representative of different network elements contained within a network, wherein respective network icons are linked together in an arrangement corresponding to how network elements are interconnected within the network and a select node configuration edit box having a user selectable vulnerability profile for selecting a vulnerability profile of a network node;

wherein selected portions of the network map turn a different color indicative of a vulnerability that has been established for that portion of the network after a security posture of the network has been established.

* * * * *

US006496209B2

(12) **United States Patent**
Horii

(10) **Patent No.: US 6,496,209 B2**
(45) **Date of Patent: *Dec. 17, 2002**

(54) **STATUS DISPLAY UNIT USING ICONS AND METHOD THEREFOR**

(75) Inventor: **Hitoshi Horii**, Tokyo (JP)

(73) Assignee: **Mitsubishi Denki Kabushiki Kaisha**, Chiyoda-Ku (JP)

(*) Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/368,577**

(22) Filed: **Aug. 5, 1999**

(65) **Prior Publication Data**

US 2002/0140725 A1 Oct. 3, 2002

(30) **Foreign Application Priority Data**

Mar. 26, 1999 (JP) 11-083052

(51) **Int. Cl.**[7] **G06F 3/00**

(52) **U.S. Cl.** **345/853**; 345/734; 345/772; 345/835; 345/837; 345/736; 345/854; 345/855

(58) **Field of Search** 345/771, 772, 345/734, 835, 837, 839, 853, 854, 855, 804, 805, 966, 969, 970, 736, 737, 738, 440; 709/223, 224

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,452,415 A * 9/1995 Hotka 395/161
5,751,931 A * 5/1998 Cox et al. 395/140
5,751,965 A * 5/1998 Mayo et al. 345/733
5,768,119 A * 6/1998 Havekost et al. 364/133
5,859,885 A * 1/1999 Rusnica et al. 376/259
5,877,766 A * 3/1999 Bates et al. 345/854
5,953,010 A * 9/1999 Kampe et al. 345/835
6,040,834 A * 3/2000 Jain et al. 345/853
6,219,046 B1 * 4/2001 Thomas et al. 345/705
6,219,050 B1 * 4/2001 Schaffer 345/781
6,243,091 B1 * 6/2001 Berstis 345/839

FOREIGN PATENT DOCUMENTS

JP 8-115199 5/1996 .... G06F/3/14
JP 10-227646 8/1998 .... G01C/21/00

* cited by examiner

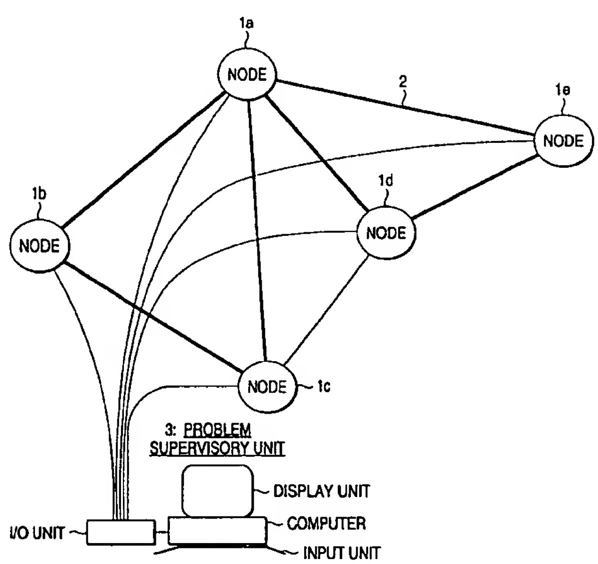
Primary Examiner—Raymond J. Bayerl
Assistant Examiner—Cuong T. Thai
(74) *Attorney, Agent, or Firm*—Burns, Doane, Swecker & Mathis, LLP

(57) **ABSTRACT**

When an abnormality detecting means **41** has detected an abnormal state of a system, an abnormal position determining means **42** determines the abnormal position in accordance with the received total address. Then, an degree-of-importance determining means determines whether or not a degree of importance exists. If the degree of importance exists, a process is performed in accordance with the degree of importance. An icon display means **45** displays a plurality of icons. A link-line display means **46** displays a link line indicating the connection between the icons. An abnormal-state alarm output means **44** determines the status to change the display in accordance with the contents of the status so as to display the changed display. The actual display is performed by a display means **48**. If necessary, output to an alarm sound producing means **49** is performed.

link
icon
status

12 Claims, 9 Drawing Sheets



flashing
color
thickness
shape
sound
size

FIG. 1

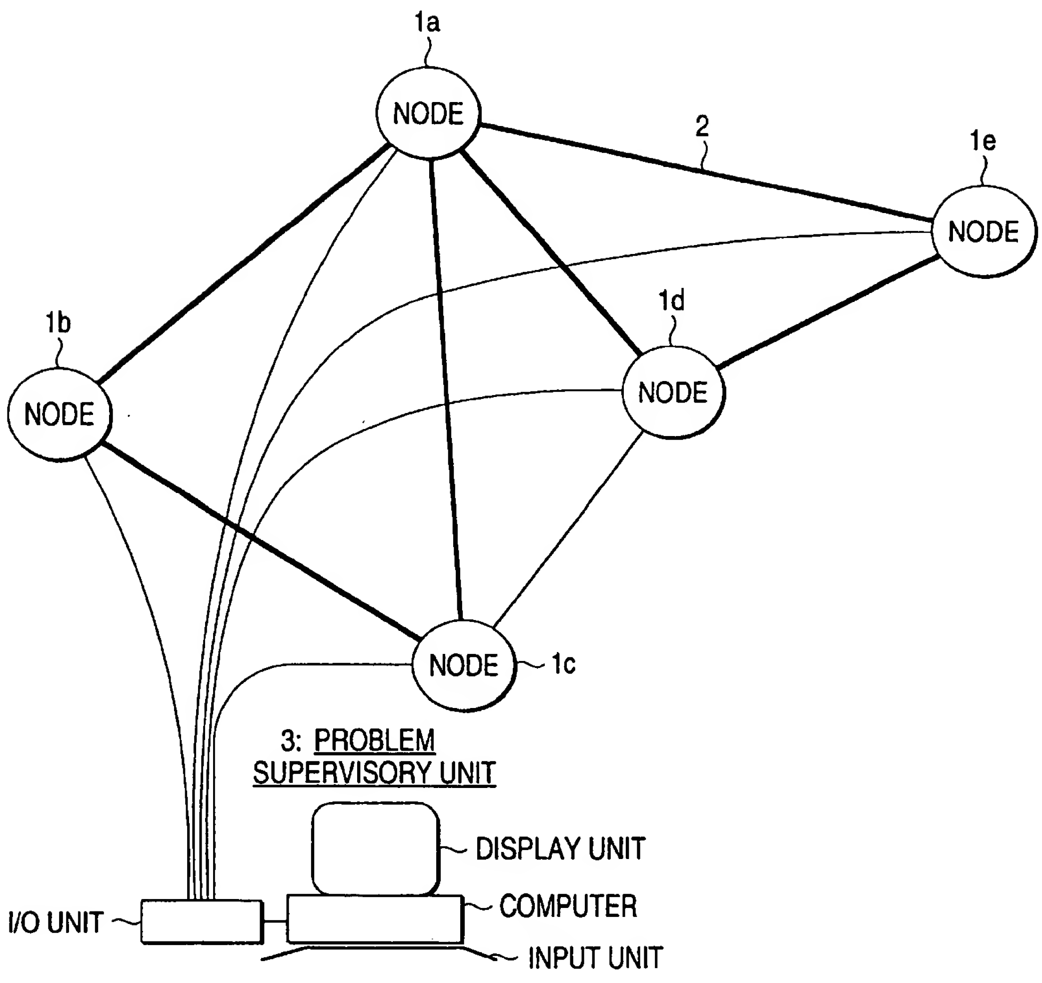

1a
NODE
2
1e
NODE
1d
NODE
1b
NODE
NODE
1c
3: PROBLEM SUPERVISORY UNIT
DISPLAY UNIT
COMPUTER
I/O UNIT
INPUT UNIT

FIG. 2

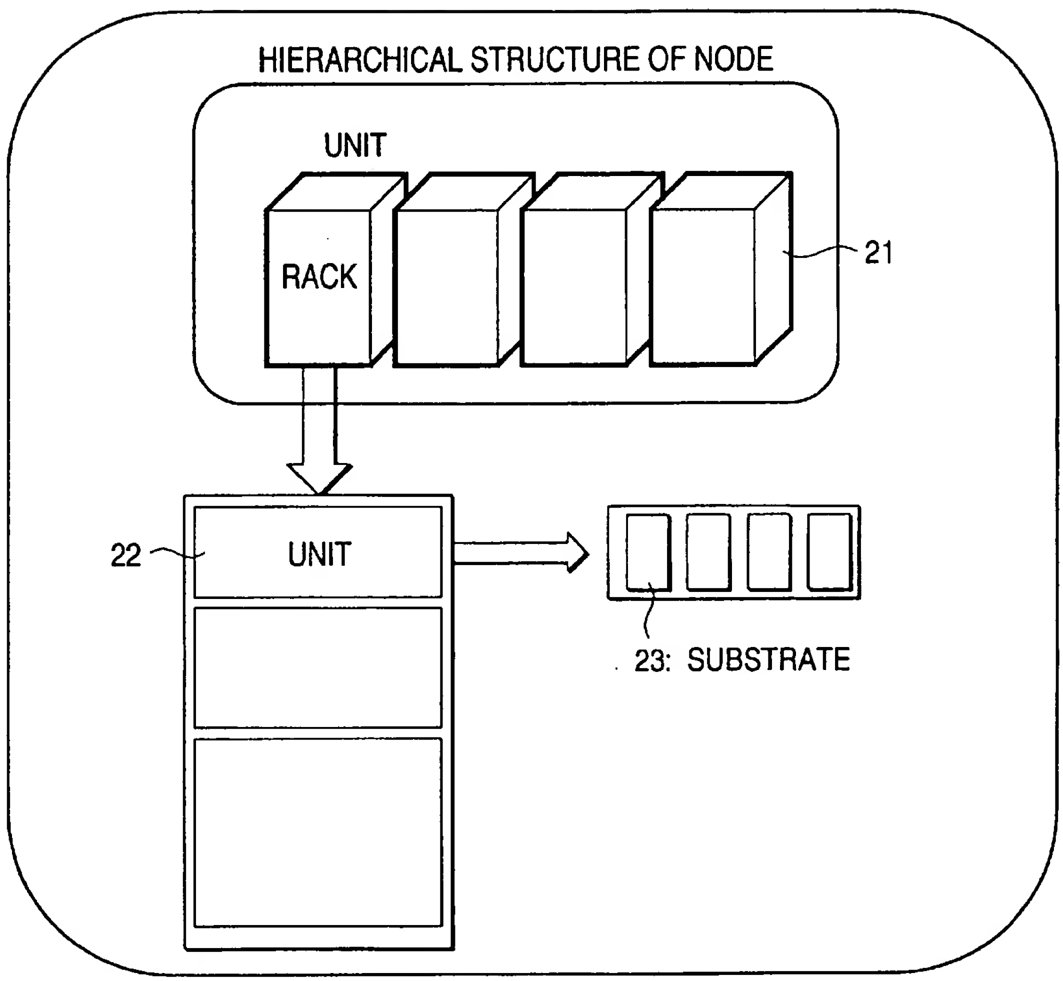

HIERARCHICAL STRUCTURE OF NODE
UNIT
RACK
21
22
UNIT
23: SUBSTRATE

FIG. 3

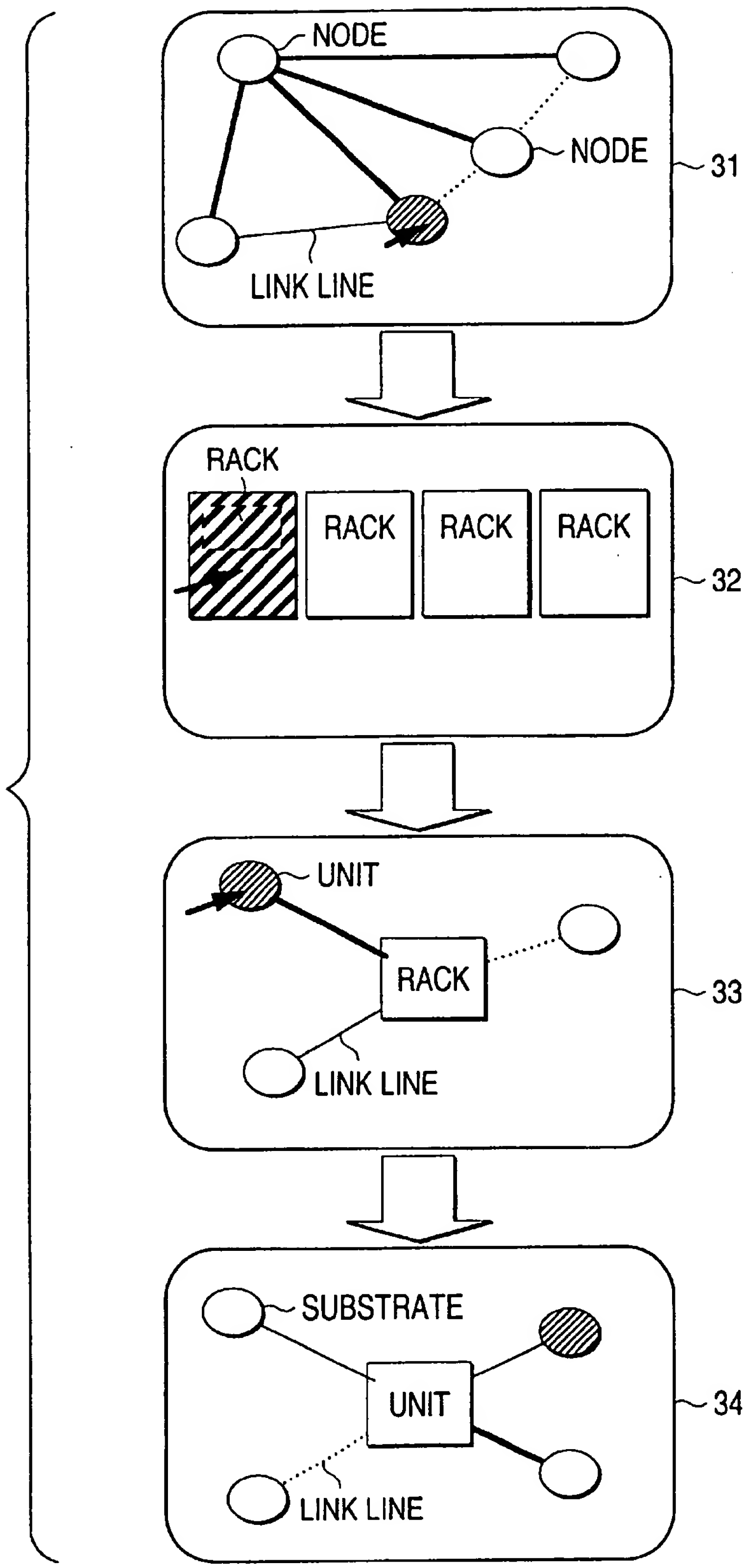

NODE
NODE
LINK LINE
31
RACK
RACK
RACK
RACK
32
UNIT
RACK
LINK LINE
33
SUBSTRATE
UNIT
LINK LINE
34

FIG. 4

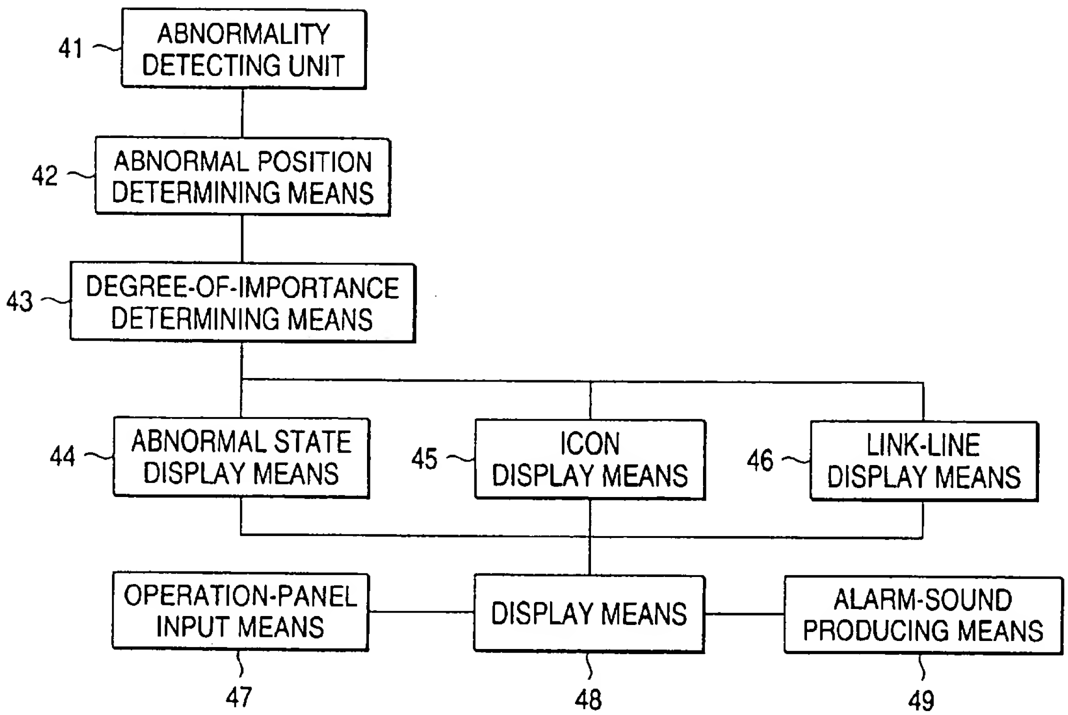

41
ABNORMALITY DETECTING UNIT
42
ABNORMAL POSITION DETERMINING MEANS
43
DEGREE-OF-IMPORTANCE DETERMINING MEANS
44
ABNORMAL STATE DISPLAY MEANS
45
ICON DISPLAY MEANS
46
LINK-LINE DISPLAY MEANS
OPERATION-PANEL INPUT MEANS
DISPLAY MEANS
ALARM-SOUND PRODUCING MEANS
47
48
49

FIG. 5

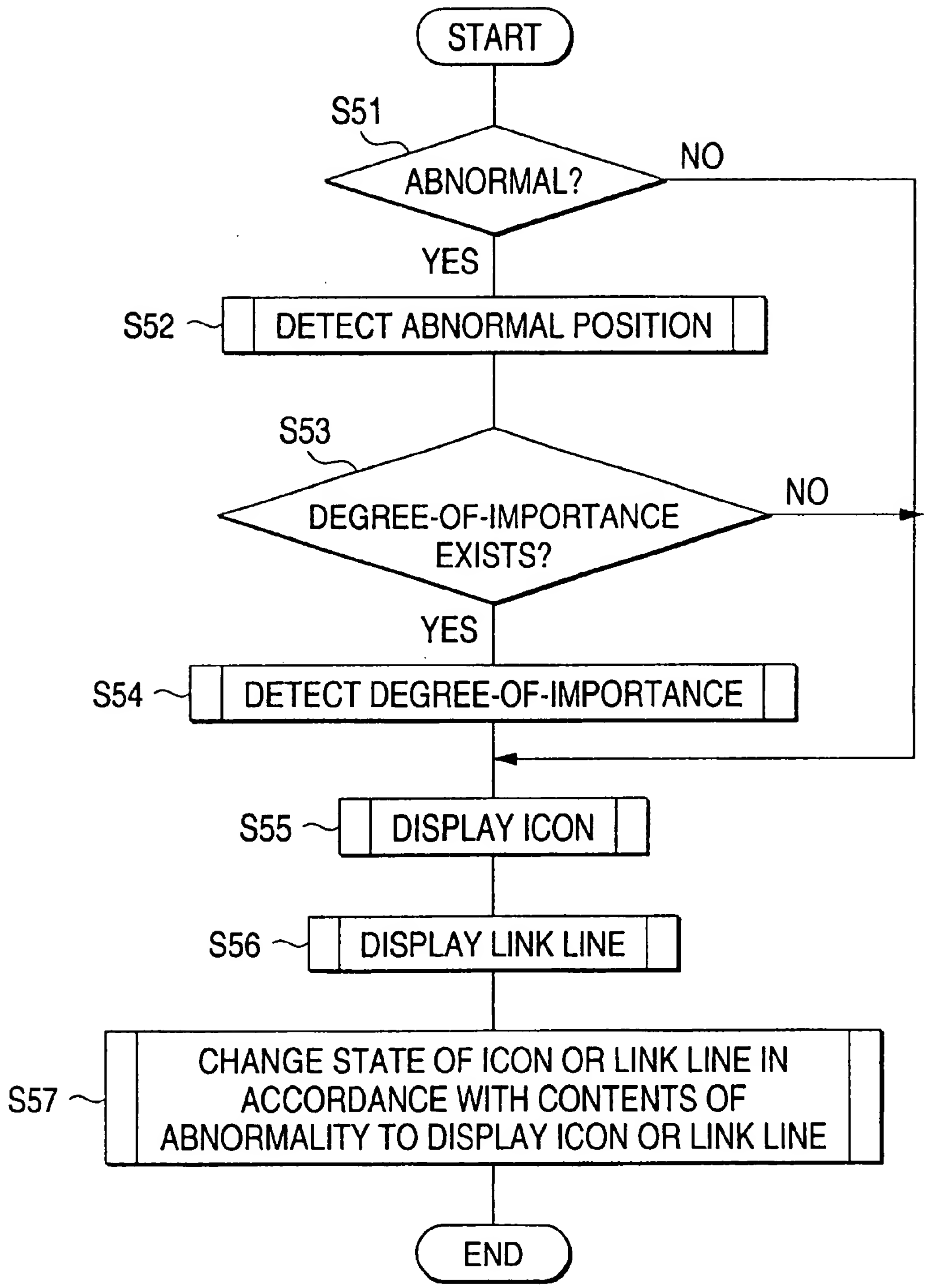

START
S51
ABNORMAL?
NO
YES
S52
DETECT ABNORMAL POSITION
S53
DEGREE-OF-IMPORTANCE EXISTS?
NO
YES
S54
DETECT DEGREE-OF-IMPORTANCE
S55
DISPLAY ICON
S56
DISPLAY LINK LINE
S57
CHANGE STATE OF ICON OR LINK LINE IN ACCORDANCE WITH CONTENTS OF ABNORMALITY TO DISPLAY ICON OR LINK LINE
END

FIG. 6

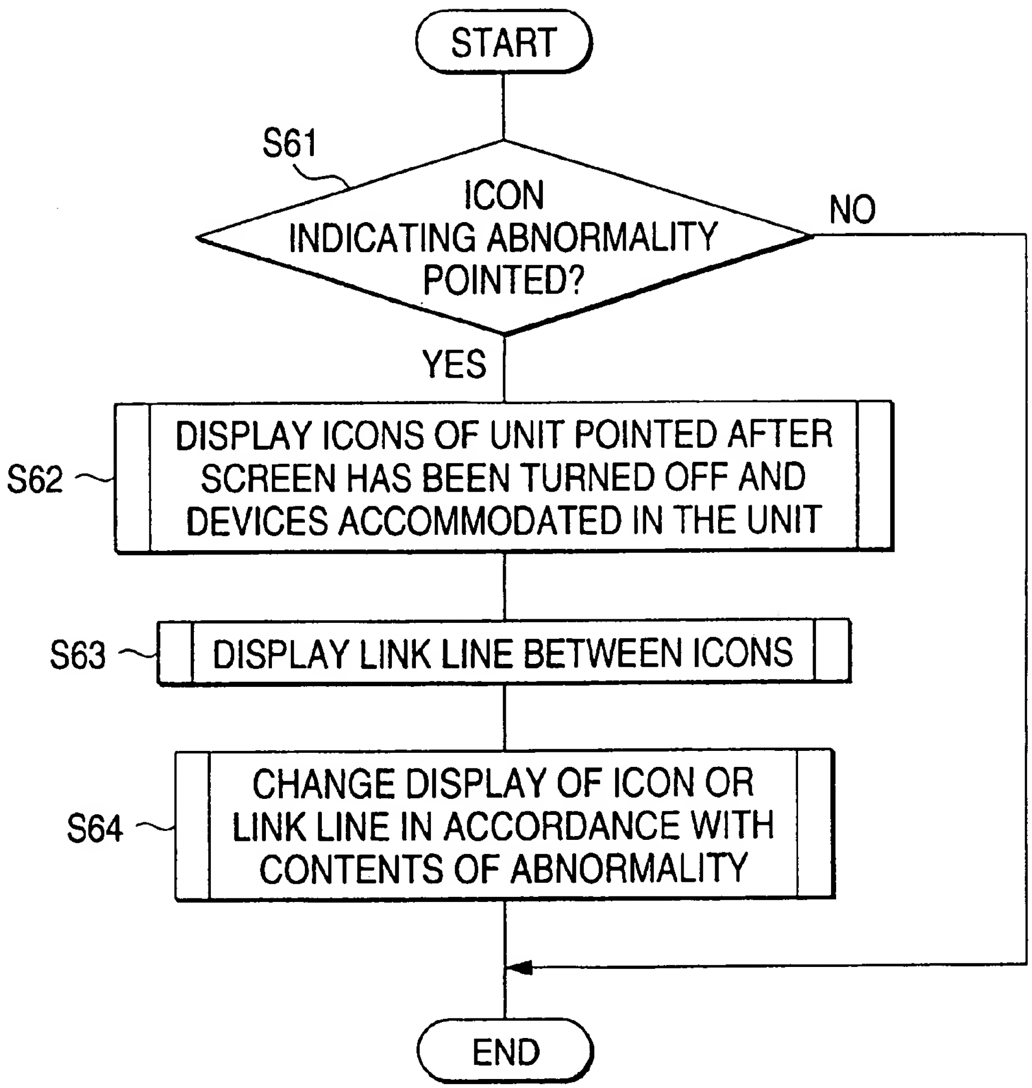

START
S61
ICON INDICATING ABNORMALITY POINTED?
NO
YES
S62
DISPLAY ICONS OF UNIT POINTED AFTER SCREEN HAS BEEN TURNED OFF AND DEVICES ACCOMMODATED IN THE UNIT
S63
DISPLAY LINK LINE BETWEEN ICONS
S64
CHANGE DISPLAY OF ICON OR LINK LINE IN ACCORDANCE WITH CONTENTS OF ABNORMALITY
END

FIG. 7

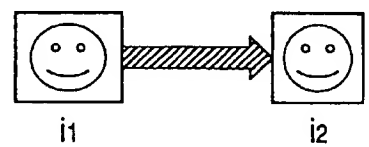
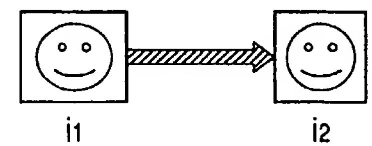
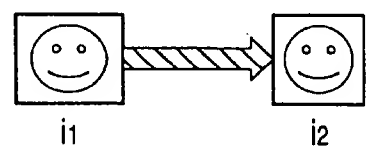
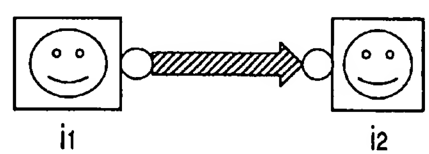
SHAPE OF LINK LINE	EXPRESSION 1	EXPRESSION 2
DIFFERENT EXPRESSION USING SHAPE, SIZE, COLOR AND DIMENSION (THREE-DIMENSIONAL SHAPE/PLANE) EXPRESSION 1 = STRONG RELATIONSHIP EXPRESSION 2 = WEAK RELATIONSHIP		
EXPRESSION USING FLASHING OF LINK LINE OR ANIMATION		
EXPRESSION USING PERIPHERAL DECORATION SYMBOLS		

FIG. 8

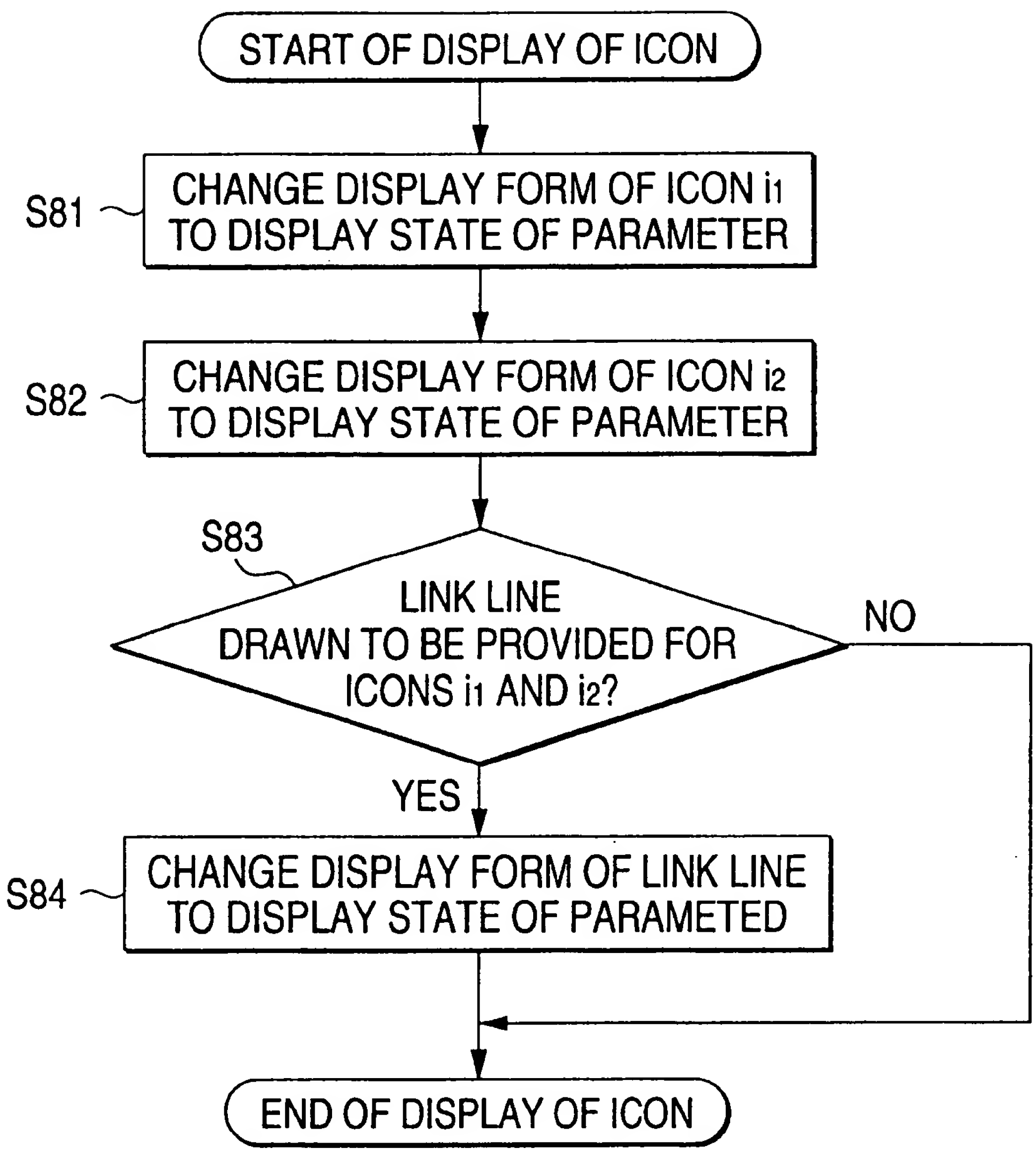
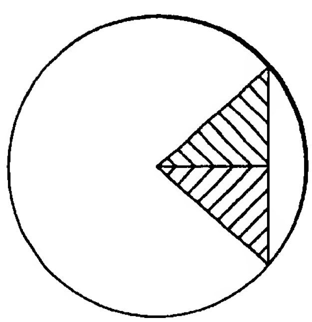
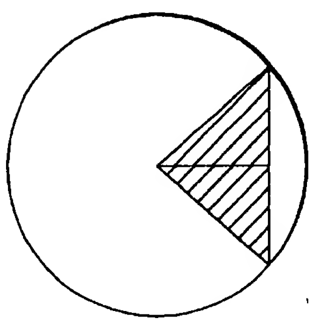

START OF DISPLAY OF ICON
S81
CHANGE DISPLAY FORM OF ICON i1 TO DISPLAY STATE OF PARAMETER
S82
CHANGE DISPLAY FORM OF ICON i2 TO DISPLAY STATE OF PARAMETER
S83
LINK LINE DRAWN TO BE PROVIDED FOR ICONS i1 AND i2?
NO
YES
S84
CHANGE DISPLAY FORM OF LINK LINE TO DISPLAY STATE OF PARAMETED
END OF DISPLAY OF ICON

FIG. 9A



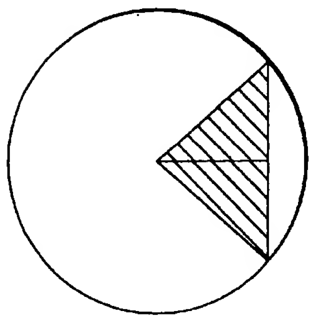
STATUS 1

FIG. 9B



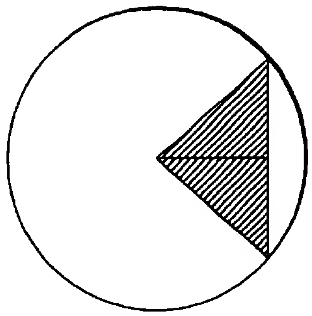
STATUS 2

FIG. 9C



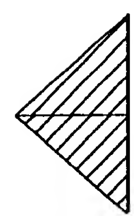
STATUS 3

FIG. 9D



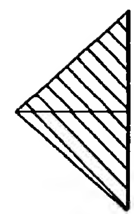
STATUS 4

FIG. 9E



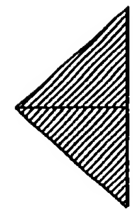
STATUS 5

FIG. 9F



STATUS 6

FIG. 9G



STATUS 7

STATUS DISPLAY UNIT USING ICONS AND METHOD THEREFOR

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a status display unit using icons for expressing changes of a variety of apparatuses among a plurality of statuses by using real-time change in the colors and shapes of icons and/or indicating relationship by using connection lines (hereinafter sometimes called "link lines") between icons and a method therefor.

Color shape

2. Description of the Related Art

A method using icons for operating application software has widely been used. For example, a method relating to display of icons has been disclosed in Japanese Patent Laid-Open No. 10-227646. FIGS. 9A to 9G show examples of icons for use in the foregoing conventional structure. The foregoing conventional structure is able to express the states of a plurality of parameters by using change in the color, the expression dimensions of planes or three-dimensional shapes or the presence or absence of peripheral decoration symbols. The disclosed structure is applied to a car navigation system. In this case, a positioning operation is performed which is a combination of the GPS navigation system for receiving an electric wave emitted from a satellite to measure the position of the vehicle and an autonomous navigation system for measuring the position of the vehicle. Then, the statuses of the positioning means based on the GPS navigation system and the positioning means based on the autonomous navigation system are classified into seven statuses below:

(1) two means are normal and the two means are being operated (status 1);

(2) two means are normal and only the positioning means based on the GPS navigation system is being operated (status 2);

(3) two means are normal and only the positioning means based on the autonomous navigation system is being operated (status 3);

(4) two means are normal and the two means are not being operated (status 4);

(5) the positioning means based on the autonomous navigation system is abnormal and only the positioning means based on the GPS navigation system is being operated (status 5);

(6) the positioning means based on the GPS navigation system is abnormal and only the positioning means based on the autonomous navigation system is being operated (status 6); and

(7) the two means are abnormal and the two means are not being operated (status 7).

Therefore, the state of the operation and whether or not each means is abnormal can easily be recognized by the user by observing the displayed screen shown in FIGS. 9A to 9G. The display is performed by using change in the color of the mark of the vehicle, the number of expression dimensions of planes or three-dimensional shapes and the presence or absence of peripheral decoration symbols (for example, a frame enclosed by a circle).

The conventional display of the status using the icon has been performed as described above. The conventional example simply relates to the method of expression of the icon. Therefore, a range which can be expressed is limited and thus there arises a problem in that a complicated state of the operation cannot easily be expressed.

There arises another problem in that the hierarchical relationship such as the parenthood and change in the status of the relationship among units belonging to a group having a given meaning of the relationship cannot be expressed.

SUMMARY OF THE INVENTION

To solve the above-mentioned problems, an object of the present invention is to provide an apparatus which enables a user to easily recognize real-time change in the status by using change in the shape and the color of a plurality of icons, presence or absence of peripheral decoration symbols and change in the expression of the relationship among the icons and a method therefor.

Shape color

A status display unit using icons according to a first aspect of the present invention is a status display unit using icons for displaying the status of a subject to be displayed by using icons or link lines each of which connects icons to each other, the status display unit using icons comprising: icon display means for displaying the icons; link line display means for displaying the link lines; status display means for changing display of a plurality of icons or link lines in accordance with the contents of the subject to be displayed so as to display the changed icons or link lines, wherein expression forms of the plural icons are changed to display the status.

appearance/content

A status display unit using icons according to a second aspect of the present invention is a status display unit using icons for use in a network supervisory apparatus, comprising: abnormal-state detecting means for detecting an abnormal state of the network; abnormal-position determining means for determining an abnormal position; icon display means for displaying icons; link-line displaying means for displaying link lines indicating the connection between icons; and abnormal-state displaying means for changing and displaying display of the icons or the link lines in accordance with the contents of the status of the network, wherein expression forms of the plural icons are changed to display the status.

A status display unit using icons according to a third aspect of the present invention has a structure that the expression forms of the plural icons are changed by changing the colors of the icons.

A status display unit using icons according to a fourth aspect of the present invention has a structure that the expression forms of the plural icons are changed by changing the number of expression dimensions of planes or three-dimensional shapes.

A status display unit using icons according to a fifth aspect of the present invention has a structure that the expression forms of the plural icons are changed by changing the presence or absence of peripheral decoration symbols.

A status display unit using icons according to a sixth aspect of the present invention is a status display unit using icons for displaying the status of a subject be displayed by using icons or link lines each of which connects icons to each other, the status display unit using icons comprising: icon display means for displaying the icons; link line display means for displaying the link lines; status display means for changing display of a plurality of icons or link lines in accordance with the contents of the subject to be displayed so as to display the changed icons or link lines, wherein expression forms of the link line are changed to display the status.

A status display unit using icons according to a seventh aspect of the present invention is a status display unit using icons for use in a network supervisory apparatus, compris-

ing: abnormal-state detecting means for detecting an abnormal state of the network; abnormal-position determining means for determining an abnormal position; icon display means for displaying icons; link-line displaying means for displaying link lines indicating the connection between icons; and abnormal-state displaying means for changing and displaying display of the icons or the link lines in accordance with the contents of the status of the network, wherein expression forms of the link line are changed to display the status.

A status display unit using icons according to an eighth aspect of the present invention has a structure that expression forms of the link lines for connecting the icons to each other are changed by changing the colors of the link lines.

A status display unit using icons according to a ninth aspect of the present invention has a structure that expression forms of the link lines for connecting the icons to each other are changed by changing the number of expression dimensions of planes or three-dimensional shapes.

A status display unit using icons according to a tenth aspect of the present invention has a structure that expression forms of the link lines for connecting the icons to each other are changed by changing the presence or absence of peripheral decoration symbols.

A status display unit using icons according to an eleventh aspect of the present invention is a status display unit using icons for displaying the status of a subject to be displayed by using icons or link lines each of which connects icons to each other, the status display unit using icons comprising: icon display means for displaying the icons; link line display means for displaying the link lines; status display means for changing display of a plurality of icons or link lines in accordance with the contents of the subject to be displayed so as to display the changed icons or link lines, wherein expression forms of the plural icons and link lines are changed to display the status.

A status display unit using icons according to a twelfth aspect of the present invention is a status display unit using icons for use in a network supervisory apparatus, comprising: abnormal-state detecting means for detecting an abnormal state of the network; abnormal-position determining means for determining an abnormal position; icon display means for displaying icons; link-line displaying means for displaying link lines indicating the connection between icons; and abnormal-state displaying means for changing and displaying display of the icons or the link lines in accordance with the contents of the status of the network, wherein expression forms of the plural icons and link lines are changed to display the status.

Icon color

A status display unit using icons according to a thirteenth aspect of the present invention has a structure that expression forms of the plural icons and link lines between the icons are changed by changing the colors of the icons and link lines.

A status display unit using icons according to a fourteenth aspect of the present invention has a structure that expression forms of the plural icons and link lines between the icons are changed by changing the number of expression dimensions of planes or three-dimensional shapes.

A status display unit using icons according to a fifteenth aspect of the present invention has a structure that expression forms of the plural icons and link lines between the icons are changed by changing the presence or absence of peripheral decoration symbols.

A status display method using icons according to a sixteenth aspect of the present invention is a status display method using icons such that the status of a subject to be displayed is displayed by using icons or link lines each of which connects icons to each other, the status display method comprising: an icon display step for displaying the icons; a link line display step for displaying the link lines; a status display step for changing display of a plurality of icons or link lines in accordance with the contents of the subject to be displayed so as to display the changed icons or link lines, wherein expression forms of the plural icons are changed to display the status.

A status display method using icons according to seventeenth aspect of the present invention is used in a network supervisory apparatus, comprising: an abnormal-state detecting step for detecting an abnormal state of the network; an abnormal-position determining step for determining an abnormal position; an icon display step for displaying icons; a link-line displaying step for displaying link lines indicating the connection between icons; and an abnormal-state displaying step for changing and displaying display of the icons or the link lines in accordance with the contents of the status of the network, wherein expression forms of the plural icons are changed to display the status.

A status display method using icons according to an eighteenth aspect of the present invention has a structure that the expression forms of the plural icons are changed by changing the colors of the icons.

A status display method using icons according to a nineteenth aspect of the present invention has a structure that the expression forms of the plural icons are changed by changing the number of expression dimensions of planes or three-dimensional shapes.

A status display method using icons according to a twentieth aspect of the present invention has a structure that the expression forms of the plural icons are changed by changing the presence or absence of peripheral decoration symbols.

A status display method using icons according to a twenty-first aspect of the present invention is a status display method using icons such that the status of a subject to be displayed is displayed by using icons or link lines each of which connects icons to each other, the status display method comprising: an icon display step for displaying the icons; a link line display step for displaying the link lines; a status display step for changing display of a plurality of icons or link lines in accordance with the contents of the subject to be displayed so as to display the changed icons or link lines, wherein expression forms of the link lines are changed to display the status.

A status display method using icons according to a twenty-second aspect of the present invention is a status display method using icons for use in a network supervisory apparatus, comprising: an abnormal-state detecting step for detecting an abnormal state of the network; an abnormal-position determining step for determining an abnormal position; an icon display step for displaying icons; a link-line displaying step for displaying link lines indicating the connection between icons; and an abnormal-state displaying step for changing and displaying display of the icons or the link lines in accordance with the contents of the status of the network, wherein expression forms of the link lines are changed to display the status.

A status display method using icons according to a twenty-third aspect of the present invention has a structure that expression forms of the link lines for connecting the icons to each other are changed by changing the colors of the link lines.

Link color

A status display method using icons according to a twenty-fourth aspect of the present invention has a structure that expression forms of the link lines for connecting the icons to each other are changed by changing the number of expression dimensions of planes or three-dimensional shapes.

A status display method using icons according to a twenty-fifth aspect of the present invention has a structure that expression forms of the link lines for connecting the icons to each other are changed by changing the presence or absence of peripheral decoration symbols.

A status display method using icons according to a twenty-sixth aspect of the present invention is a status display method using icons such that the status of a subject to be displayed is displayed by using icons or link lines each of which connects icons to each other, the status display method comprising: an icon display step for displaying the icons; a link line display step for displaying the link lines; a status display step for changing display of a plurality of icons or link lines in accordance with the contents of the subject to be displayed so as to display the changed icons or link lines, wherein expression forms of the plural icons and link lines are changed to display the status.

icon, link

A status display method using icons according to a twenty-seventh aspect of the present invention is a status display method using icons for use in a network supervisory apparatus, comprising: an abnormal-state detecting step for detecting an abnormal state of the network; an abnormal-position determining step for determining an abnormal position; an icon display step for displaying icons; a link-line displaying step for displaying link lines indicating the connection between icons; and an abnormal-state displaying step for changing and displaying display of the icons or the link lines in accordance with the contents of the status of the network, wherein expression forms of the plural icons and link lines are changed to display the status.

A status display method using icons according to a twenty-eighth aspect of the present invention has a structure that expression forms of the plural icons and link lines between the icons are changed by changing the colors of the icons and link lines.

color of link icon and

A status display method using icons according to a twenty-ninth aspect of the present invention has a structure that expression forms of the plural icons and link lines between the icons are changed by changing the number of expression dimensions of planes or three-dimensional shapes.

A status display method using icons according to a thirtieth aspect of the present invention has a structure that expression forms of the plural icons and link lines between the icons are changed by changing the presence or absence of peripheral decoration symbols.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a view showing an example of the structure of a network;

FIG. 2 is a view showing the structure of an example in which one or more racks accommodated in each node shown in FIG. 1 are displayed on a screen;

FIG. 3 is a diagram showing an example of a displayed screen of a problem supervisory unit;

FIG. 4 is a diagram showing an example of the structure for realizing the operation of the problem supervisory unit;

FIG. 5 is a flow chart showing the process which is performed by the problem supervisory unit 3 to realize the operation according to the first embodiment;

FIG. 6 is a flow chart showing the operation of the problem supervisory unit according to the embodiment which is performed when a user has pointed an icon encountered a problem;

FIG. 7 is a diagram showing an example of a screen which specifically displays icons and the relationship between the icons;

FIG. 8 is a flow chart showing the process for specifically displaying the icons and the relationship between the icons; and

FIGS. 9A to 9G are diagrams showing examples of icons for use in a conventional structure.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

A description will be given in more detail of preferred embodiments of the present invention with reference to the accompanying drawings.

(First Embodiment)

FIG. 1 is a schematic view showing an example of the structure of a network. Referring to the drawing, reference numeral **1** represents nodes, **2** represents connection lines for establishing the connection between the nodes, the connection lines corresponding to optical cables for use in a large-size network system. Reference numeral **3** represents a problem supervisory unit for supervising whether or not the network has a problem. The problem supervisory unit **3** is exemplified by a computer incorporating an input unit, an I/O unit and a display unit, as shown in the drawing.

FIG. 2 is a view showing an example of the structure in which one or more cabinets (hereinafter called "racks") accommodated in each node shown in FIG. 1. Each rack accommodates one or more units. Each unit accommodates one or more circuit substrates (hereinafter simply called "substrates").

FIG. 3 is a diagram showing an example of a display screen of the problem supervisory unit, in which detection of a position of occurrence of a problem in the network is illustrated. In the drawing, supervisory of change in the icons and links between the icons is performed.

The operation of the network supervisory system will now be described.

In the network shown in FIG. 1, an assumption is made that a node address, a rack address, a unit address and a substrate address are previously assigned to each of the nodes, racks, units and the substrates. Moreover, a total address which is the logical sum of the node address, the rack address, the unit address and the substrate address is assigned to be only address in the network.

When a certain substrate (for example, substrated of unit a of rack a of node 1c) in the node 1c connected to the network has encountered an abnormal condition, the node **1c** accommodating the foregoing substrate detects an abnormal signal and sequentially reads rack address a, unit address a and substrate address d which have encountered the abnormality. Then, the address **1c** of the node is added to the rack address a, unit address a and the substrate address d which has encountered the abnormality so that address "1caad" is produced. The address "1caad" is, together with an abnormal signal and an abnormal status information indicating a type of the abnormality, communicated to the problem supervisory unit **3**.

When the problem supervisory unit **3** has detected the abnormal signal and abnormal status information, the problem supervisory unit **3** divides "1caad" into the address **1c** of the node encountered the abnormality, the rack address a,

Sound Red

the unit address a and the substrate address d to manage the divided addresses. Moreover, the problem supervisory unit 3 stores the abnormal status. In addition, occurrence of the abnormality is alarmed by operating an alarm unit, such as a buzzer and by changing the color of the icon of the node 1*a* displayed on the display unit to, for example, red.

Then, the user stops the sound production from the alarm unit by operating a confirmation button or the like, and then points the icon of the node 1*c* which is indicating the abnormality on the screen by using a pointing device (or by directly touching a touch panel in the case of the touch panel). Thus, the network and icons of all of nodes disappear from the screen. As an alternative to this, icons of the plurality of the racks a to d accommodated in the node 1*c* encountered the abnormality are displayed. Moreover, the color of the icon of the rack a among the foregoing racks encountered the abnormality is changed to red. In addition, the thickness of the link line between the icon of the node 1*c* and that of the rack a is changed to a thick line.

thickness

Then, the user uses the pointing device to point the icon of the rack a indicating the abnormality on the screen. Thus, the screen is switched so that the icon of the rack a encountered the abnormality and icons of the plural units a to c accommodated in the foregoing rack are displayed in a state in which the displayed icons are connected to one another with link lines. Moreover, the color of the icon of the unit a among the foregoing units encountered the abnormality is changed to red (a green icon is displayed in a normal state). Moreover, the thickness of the link line between the icon of the rack a and the icon a of the unit a is changed to a thicker line.

Red, green thickness

Then, when the user has used the pointing device to point the icon of the unit a indicating the abnormality on the screen, the screen is switched. Thus, the icon of the unit a encountered the abnormality and the icons of the plural substrates a to d accommodate in the foregoing unit are displayed such that the displayed icons are connected to one another with the link lines. Moreover, the color of the icon of the substrate d among the displayed substrates which has encountered the abnormality is changed to a color corresponding to each of the contents of the abnormal status (for example, yellow is used in a case of defective connection and red is used in a case of a failure of the circuit) Moreover, the thickness of the link line between the icon of the unit a and that of the substrate d is changed to a thicker line.

The foregoing display of the abnormality is reset when the user depresses a restoring button so that an initial screen (a screen 31 in which icons of all nodes are displayed in green) is again displayed.

The example of the applied system is structured as described above.

Although the foregoing embodiment is arranged such that the color is changed to red, another color may, of course, be employed. Moreover, sound production using the alarm unit may be omitted.

Although the color of the icon is changed in the foregoing embodiment, the present invention is not limited to this. For example, change in the display size of the icon to smaller size and larger size may be repeated. As an alternative to this, the icon may be flashed or the shape of the icon may be changed to another shape. As an alternative to this, the thickness of the frame may be changed.

Size & shape flashing link color thickness thinner

In this embodiment, the thickness of the link line between the icon is changed. Note that the present invention is not limited to this. For example, the color of the link line may be changed. As an alternative to this, the operation for changing the thickness of the connection line to a larger thickness or smaller thickness may be repeated. As an alternative to this, the connection line may be flashed. As an alternative to this, the shape of the connection line may be changed to another shape (for example, an alternate long and short dash line or a line having black circles or black triangles). As an alternative to this, the thickness of the frame may be changed.

flashing

Although the foregoing embodiment employs the network as an example, the present invention is not limited to this. The present invention may be applied to any one of systems and apparatuses to which the spirit of the display of the status using the icon can be applied.

(Second Embodiment)

FIG. 4 is a diagram showing an example of the structure for permitting the problem supervisory unit to perform the above-mentioned operation. Referring to FIG. 4, reference numeral **41** represents an abnormality detecting means, **42** represents an abnormal position determining means, **43** represents a degree-of-importance determining means, **44** represents an abnormal-state alarm output means, **45** represents an icon display means, **46** represents a link-line display means, **47** represents an operation-panel input means which accommodates a variety of buttons, **48** represents a display means and **49** represents an alarm sound producing means.

The operation of each means will now be described. Referring to FIG. 4, when the abnormality detecting means **41** has detected the abnormal signal, the abnormality detecting means **41** outputs, to the abnormal position determining means **42**, the total address which is the logical sum of the node address, the rack address, the unit address and the substrate address in the portion encountered the abnormality. The abnormal position determining means **42** divides the supplied total address into the node address, the rack address, the unit address and the substrate address to determine the position at which the abnormality has occurred. If a plurality of abnormal states occur simultaneously, the degree-of-importance determining means **43** makes a reference to an included table (not shown) to determine the degree of importance to select the abnormality having a higher degree of importance. The abnormal-state alarm output means **44** uses the abnormal status communicated from the degree-of-importance determining means **43** as a key in a process for retrieving an included abnormal status table (not shown). Thus, the abnormal-state alarm output means **44** decides the corresponding type of the display (for example, display in yellow is selected when the status is "caution" and display in red is selected when the status is "failure"). Simultaneously, the abnormal-state alarm output means **44** instructs the alarm sound producing means **49** to make an output.

Then, the icon display means **45** instructs the display means **48** to display the icon. The link-line display means **46** instructs the display means **48** to connect the icons to each other with the link line. Specifically, the display is performed by using a known development tool for the window. The abnormal-state alarm output means **44** instructs the display means **48** to display the decided color such that the color is superimposed on the icon.

The display means **48** follows the instructions issued from the abnormal-state alarm output means **44**, the icon display means **45** and the link-line display means **46** to perform display. The alarm sound producing means **49** follows an instruction issued from the abnormal-state alarm output means **44** to perform the alarming operation. When the user depresses the confirmation button on the operation-panel input means **47**, the alarm sound producing means receives the signal to stop producing the sound. When the user has

depressed the restoring button on the operation-panel input means **47**, the abnormal-state alarm output means **44** cancels the color display of the abnormal state under the condition that the abnormal state has been restored. The icon display means **45** displays the icon of the initial screen. Also the link-line display means **46** displays the link line between the icons on the initial screen.

(Third Embodiment)

FIG. **5** is a flow chart showing the process for realizing the operation of the problem supervisory unit according to the first embodiment.

Referring to FIG. **5**, the foregoing process is periodically performed. Note that the foregoing process is sometime started manually, if necessary. If the process has been started, whether or not an abnormal state has occurred is detected (step **S51**). If no abnormal state is detected, the operation proceeds to step **S55**. If an abnormal state is detected in step **S51**, the received total address is divided into the node address, the rack address, the unit address and the substrate address to determine the abnormal position (step **S52**). Then, existence of the degree of importance is determined to determine whether or not an important abnormal state has occurred (step **S53**). If the degree of importance does not exist in step **S53**, the operation proceeds to step **S55**. If the degree of importance exists, the degree of importance is detected. If two or more abnormal states must simultaneously be overcome, the abnormal state having the higher degree of importance is selected (step **S54**). Then, the icon is displayed (step **S55**). Then, the link line for connecting the icons to each other is displayed (step **S56**). Then, the abnormal status is examined (this process is omitted from illustration) In accordance with the contents of the abnormal state, the state of the icon or the link line is changed and the changed icon or the link line is displayed (step **S57**).

FIG. **6** is a flow chart of the operation of the problem supervisory unit according to this embodiment which is performed when the user has pointed an icon having a problem.

The operation shown in FIG. **6** will now be described.

Referring to FIG. **6**, the problem supervisory unit checks whether or not the icon having abnormality has been pointed by the user (step **S61**). If the pointing operation is not performed, the foregoing process is completed. If the pointing operation is detected in step **S61**, the screen which is being displayed at present is turned off. Then, the pointed icon of the unit and the icons of units accommodated in the foregoing unit are displayed (step **S62**). Moreover, the link lines for establishing the connection between icons are displayed (step **S63**). Then, the abnormal status is examined (this process is omitted from description). In accordance with the contents of the abnormality, the color of the icon or the link line to be displayed is changed (step **S64**).

(Fourth Embodiment)

FIG. **7** is a diagram showing an example of a screen for specifically displaying icons and the relationship between icons. As shown in the drawing, the expression of the connection line for connecting icons to each other in a case where two icons exist is changed. In an example case shown in the uppermost portion of the drawing, existence or absence of the subordination between the icons and the degree of coupling between the icons are expressed by using the shape, the size and the color of the link line and the different manner of expression using a plane or a three-dimensional shape. Since an arrow in Expression 1 is a thick arrow, a strong degree of coupling is indicated. Since an arrow in Expression 2 is a thin arrow, a weak degree of coupling is indicated.

thick/thin

The intermediate example is arranged to perform expression using flashing of the link line and animation. In the lower case, peripheral symbols of the link line are used to perform the expression.

FIG. **8** is a flow chart of a process for specifically displaying the icons and the relationship between the icons. The process of the foregoing flow chart will now be described. Referring to FIG. **8**, the state of display of icon I_1 is changed by using the color, the number of expression dimensions of planes or three-dimensional shapes and peripheral decoration symbols. Thus, the states of a plurality of parameters are displayed (step **S81**).

Then, the state of display of icon I_2 is changed by using the color, the number of expression dimensions of planes or three-dimensional shapes and peripheral decoration symbols. Thus, the states of a plurality of parameters are displayed (step **S82**).

Then, whether or not the icon I_1 and the icon I_2 have a relationship is examined (step **S83**) If the icons I_1 and I_2 have no relationship, the process is completed. If the relationship exists, the link line between the two icons is displayed (step **S84**).

Then, the state of display of the link line between the icon I_1 and the icon I_2 is changed by using the color, the number of expression dimensions of planes or three-dimensional shapes and peripheral decoration symbols. Thus, the states of a plurality of parameters are displayed (step **S85**).

According to this embodiment, real-time change in the status is visually displayed by using the number of expression dimensions of planes or three-dimensional shapes and peripheral decoration symbols. Therefore, the range in which the icon can be expressed can be enlarged and, therefore, a complicated state of the operation can easily be expressed.

According to this embodiment, the hierarchical relationship such as the parenthood and change in the status of the relationship among units belonging to a group having a given meaning of the relationship can easily be expressed by performing visual display using change in the color and the shape of the connection line for establishing the connection between the icons.

As described above, according to the present invention, real-time change in the status is visually displayed by using the number of expression dimensions of planes or three-dimensional shapes and peripheral decoration symbols. Therefore, an effect can be obtained in that the range in which the icon can be expressed can be enlarged and, therefore, a complicated state of the operation can easily be expressed.

According to the present invention, an effect can be obtained in that the hierarchical relationship such as the parenthood and change in the status of the relationship among units belonging to a group having a given meaning of the relationship can easily be expressed by performing visual display using change in the color and the shape of the connection line for establishing the connection between the icons.

What is claimed is:

1. A status display unit using icons for displaying the status of a subject which is displayed by using icons or link lines each of which connects icons to each other, said status display unit using icons comprising:

icon display means for displaying said icons;

link line display means for displaying said link lines;

degree-of-importance determining means for determining a degree of importance of a contents of the subject; and

status display means for changing display of a plurality of icons or link lines in accordance with the degree of

importance of the contents of the subject to be displayed so as to display the icons or link lines having a higher degree of importance;

wherein expression forms of the plural icons are changed to display the status.

2. A status display unit using icons for use in a network supervisory apparatus, comprising:

abnormal-state detecting means for detecting an abnormal state of the network;

abnormal-position determining means for determining an abnormal position;

degree-of-importance determining means for determining a degree of importance of a contents of the status of the network;

icon display means for displaying icons;

link-line displaying means for displaying link lines indicating the connection between icons; and

abnormal-state displaying means for changing and displaying a higher degree of importance of the icons or the link lines in accordance with the degree of importance of the contents of the status of the network;

wherein expression forms of the plural icons are changed to display the status.

3. A status display unit using icons for displaying the status of a subject to be displayed by using icons or link lines each of which connects icons to each other, said status display unit using icons comprising:

icon display means for displaying said icons;

link line display means for displaying said link lines;

degree-of-importance determining means for determining a degree of importance of a contents of the subject; and

status display means for changing display of a plurality of icons or link lines in accordance with the degree of importance of the contents of the subject to be displayed so as to display the icons or link lines having a higher degree of importance;

wherein expression forms of the link line are changed to display the status.

4. A status display unit using icons for use in a network supervisory apparatus, comprising:

abnormal-state detecting means for detecting an abnormal state of the network;

abnormal-position determining means for determining an abnormal position;

degree-of-importance determining means for determining a degree of importance of a contents of the status of the network;

icon display means for displaying icons;

link-line displaying means for displaying link lines indicating the connection between icons; and

abnormal-state displaying means for changing and displaying a higher degree of importance of the icons or the link lines in accordance with the degree of importance of the contents of the status of the network,

wherein expression forms of the link line are changed to display the status.

5. A status display unit using icons for displaying the status of a subject to be displayed by using icons or link lines each of which connects icons to each other, said status display unit using icons comprising:

icon display means for displaying said icons;

link line display means for displaying said link lines;

degree-of-importance determining means for determining a degree of importance of a contents of the subject; and

status display means for changing display of a plurality of icons or link lines in accordance with the degree of importance of the contents of the subject to be displayed so as to display the icons or link lines having a higher degree of importance;

wherein expression forms of the plural icons and link lines are changed to display the status.

6. A status display unit using icons for use in a network supervisory apparatus, comprising:

abnormal-state detecting means for detecting an abnormal state of the network;

abnormal-position determining means for determining an abnormal position;

degree-of-importance determining means for determining a degree of importance of a contents of the status of the network;

icon display means for displaying icons;

link-line displaying means for displaying link lines indicating the connection between icons; and

abnormal-state displaying means for changing and displaying a higher degree of importance of the icons or the link lines in accordance with the degree of importance of the contents of the status of the network;

wherein expression forms of the plural icons and link lines are changed to display the status.

7. A status display method using icons such that the status of a subject to be displayed is displayed by using icons or link lines each of which connects icons to each other, said status display method comprising the steps of:

displaying the icons;

displaying said link lines;

determining a degree of importance of a contents of the subject; and

changing display of a plurality of icons or link lines in accordance with the degree of importance of the contents of the subject to be displayed so as to display the icons or link lines having a higher degree of importance;

wherein expression forms of the plural icons are changed to display the status.

8. A status display method using icons for use in a network supervisory apparatus, said method comprising the steps of:

detecting an abnormal state of the network;

an abnormal-position determining step for determining an abnormal position;

determining a degree of importance of a contents of the status;

displaying icons;

displaying link lines indicating the connection between icons; and

changing and displaying a higher degree of importance of the icons or the link lines in accordance with the degree of importance of the contents of the status of the network;

wherein expression forms of the plural icons are changed to display the status.

9. A status display method using icons such that the status of a subject to be displayed is displayed by using icons or link lines each of which connects icons to each other, said status display method comprising the steps of:

displaying the icons;

displaying said link lines;

determining a degree of importance of a contents of the subject; and

changing display of a plurality of icons or link lines in accordance with the degree of importance of the contents of the subject to be displayed so as to display the icons or link lines having a higher degree of importance;

wherein expression forms of the link lines are changed to display the status.

10. A status display method using icons for use in a network supervisory apparatus, said method comprising the steps of:

detecting an abnormal state of the network;

an abnormal-position determining step for determining an abnormal position;

determining a degree of importance of a contents of the status;

displaying icons;

displaying link lines indicating the connection between icons; and

changing and displaying a higher degree of importance of the icons or the link lines in accordance with the degree of importance of the contents of the status of the network;

wherein expression forms of the link lines are changed to display the status.

11. A status display method using icons such that the status of a subject to be displayed is displayed by using icons or link lines each of which connects icons to each other, said status display method comprising the steps of:

determining a degree of importance of a contents of the subject;

displaying the icons;

displaying said link lines;

changing display of a plurality of icons or link lines in accordance with the degree of importance of the contents of the subject to be displayed so as to display the icons or link lines having a higher degree of importance;

wherein expression forms of the plural icons and link lines are changed to display the status.

12. A status display method using icons for use in a network supervisory apparatus, said method comprising:

detecting an abnormal state of the network;

determining an abnormal position;

determining a degree of importance of a contents of the status;

displaying icons;

displaying link lines indicating the connection between icons; and

changing and displaying a higher degree of importance of the icons or the link lines in accordance with the degree of importance of the contents of the status of the network;

wherein expression forms of the plural icons and link lines are changed to display the status.

* * * * *